

**TRANSCRIPTOMIC INVESTIGATION OF NANOPARTICLE
TOXICITY ON HepaRG SPHEROIDS**

**A DISSERTATION SUBMITTED TO
THE GRADUATE SCHOOL OF ENGINEERING AND SCIENCE
OF BILKENT UNIVERSITY
IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR
THE DEGREE OF
DOCTOR OF PHILOSOPHY
IN
MATERIALS SCIENCE AND NANOTECHNOLOGY**

By

Merve Erden

January 2025

TRANSCRIPTOMIC INVESTIGATION OF NANOPARTICLE TOXICITY ON HepaRG SPHEROIDS

By Merve ERDEN

January 2025

We certify that we have read this dissertation and that in our opinion it is fully adequate, in scope and in quality, as a thesis for the degree of Doctor of Philosophy.

Urartu Özgür Şafak Şeker (Advisor)

Çağdaş Devrim Son

Bahar Değirmenci Uzun

Özlen Konu Karakayalı

Nihal Aydoğan

Approved for the Graduate School of Engineering and Science:

Orhan Arıkan

Director of the Graduate School

ABSTRACT

TRANSCRIPTOMIC INVESTIGATION OF NANOPARTICLE TOXICITY ON HepaRG SPHEROIDS

Merve Erden

PhD in Materials Science and Nanotechnology

Advisor: Urartu Özgür Şafak Şeker

January 2025

Nanoparticles (NPs) are used in diverse fields such as cosmetics, drug delivery, food industry and medicine. They are easy to produce and easy to modify according to the needs of the application. However, there is an arising concern about the adverse effects of the nanoparticles. Negative impacts of the nanoparticles on human body needs further analysis because nanoparticle exposure of human body increases every day with the widespread usage of these materials in daily products. In literature, the studies about the nanoparticle toxicity remain inadequate since they generally use insufficient number and non-variety of nanoparticles, inappropriate cell line or cell culture method. This study focuses on evaluation of nanoparticle toxicity with transcriptomic analysis. With the help of these analysis, we aim to find the genes that respond to NP toxicity, which might help us better understand the mechanism of the toxicity. HepaRG cells were used as the best alternative to primary hepatocytes. Additionally, to get the results most similar to the in

vivo liver organ, HepaRG cells were cultured in 3D spheroids then they were incubated with eleven different nanoparticles of varying materials and sizes. To reveal the genes that responds to nanoparticle toxicity and to find the toxicity mechanisms of nanoparticles on HepaRG spheroids, transcriptome sequencing (RNA-seq) was conducted. Anticipatedly, cellular processes such as drug metabolism, sensitivity to metal ions, oxidative stress, EMT and apoptosis pathways showed enrichment in the DEGs of all nanoparticles. The common and unique genes inbetween different types and sizes of NPs were noted. These results offer novel findings about the toxicity mechanisms of HepaRG spheroids in response to NPs.

Keywords: nanomaterials, nanotoxicity, synthetic biology, nanoparticles, transcriptomics, toxicity, nanotechnology, bioinformatics

ÖZET

NANOPARÇACIKLARIN HepaRG SFEROİDLERİ ÜZERİNDEKİ TOKSİSİTESİNİN TRANSKRİPTOMİK OLARAK İNCELENMESİ

Merve Erden

Malzeme Bilimi Ve Nanoteknoloji, Doktora

Tez Danışmanı: Urartu Özgür Şafak Şeker

Ocak 2025

Nanoparçacıklar kozmetik, ilaç dağıtımı ve tıp gibi birçok alanda kullanımı olmasına rağmen birçok deavantajı da vardır. Nanoparçacıkların zararlı etkileri henüz tam anlamıyla anlaşılamamıştır ve daha ileri araştırmalara ihtiyaç vardır. Literatürde nanoparçacıklar üzerinde yapılan çalışmalar ya az sayıda nanoparçacıkla veya uygun olmayan hücre hattının veya kültür yönteminin kullanımıyla sınırlı kalmıştır. Bu çalışma nanoparçacıkların HepaRG hücreleri üzerindeki toksisitesinin transkriptomik olarak incelenmesi ve toksisiteye cevap veren genlerin tespit edilmesi üzerine odaklanmaktadır. Primer karaciğer hücresine en iyi alternatif olarak HepaRG hücreleri seçilmiştir. HepaRG hücreleri karaciğerde bulunan hücrelerin mikroçevresini daha iyi oluşturabilmek için 3 boyutlu olarak kültür edilmiştir. Daha sonra oluşturulan 3 boyutlu hücre sferoidleri farklı boyuta ve materyala sahip 11 nanoparçacıkla muamele edilmiştir. Nanoparçacıkların HepaRG hücreleri üzerindeki toksisite mekanizmasının anlaşılması için transkriptomik sekanslama analizleri yapılmıştır. Sonuç olarak, ilaç metabolizması, metal iyon metabolizması, oksidatif stres, epitel-mezenkimal dönüşüm (EMT) ve apoptoz gibi hücre mekanizmalarının ifadesi tüm nanoparçacıklar arasında ortak olarak artmıştır.

Nanoparçacık toksisitesine yanıt veren ortak ve özgün genler belirtilmiştir. Bu sonuçlar HepaRG hücrelerinin nanoparçacık toksisitesine verdiği yanıtın mekanizmasının aydınlatılmasına yeni bilgiler eklemektedir.



Anahtar Kelimeler: nanomateryal, nanotoksosite, sentetik biyoloji, nanoparçacık, transcriptom analizi, toksisite, nanoteknoloji, biyoinformatik

ACKNOWLEDGEMENTS

I would like to express my deepest gratitude and warmest thanks to my advisor and mentor, Dr. Urartu Özgür Şafak Şeker. Doing a PhD is a long journey and in this kind of journey you need a person who will always be there to support and guide you. Dr. Urartu was the most kind and supportive supervisor in this hard journey. He always motivated and guided me in the process of learning new things. He was always there whenever I had an obstacle to pass through. I was always inspired with his research passion, and I am very thankful and happy to complete my PhD in his lab. It is a privilege to work under his guidance. Without his support and understanding this dissertation would not be possible.

I would like to thank to Prof. Çağdaş Devrim Son and Dr. Bahar Değirmenci Uzun for being my thesis tracking committee member for all these years. I am grateful for your valuable advice, ideas and comments. I owe my special thanks to Dr. Özlen Konu for her contribution in this project with her ideas and guidance as well as being a part of my defense jury member. Also, I would like to thank to Dr. Nihal Aydoğan for accepting to be my jury member for my thesis defense and for your valuable time.

I would like to thank to my husband Suat Tüçer and to Nazlıcan Tunç for their support and contribution in this project while I was out. Being part of a friendly lab group is a luck. I want to thank to all former and current SBL group members; Abdurrahman Atila, Ahmet Hınçer, Aslı Semerci, Cemile Elif Özçelik, Doğu Akboğa, Ece Avcı, Elif Duman, Mehmet Emin Bakar, Melis Karaca, Merve Yavuz, Nazlıcan Tunç, Recep Erdem Ahan, Senem Şen, Cem Dirse Polat, Derin Akman, Aslı Doğruer, Damla Albayrak and Fayiti Fuerkaiti for their friendship and warm conversations. You all supported me when I felt

tired and crushed with the responsibilities of PhD and motherhood. I owe my special thanks to Gizem Makas for being my free psychological counselor. She always relieved me with her experience. I would like to thank to Elif Duman for her advice on motherhood. I would like to also thank to Melis for caring for my daughter every time she comes to UNAM.

I would like to thank to my family, my mother, my father, my sister and my brothers for their support. I am very lucky to have them; they give me the reliance that whatever happens to me they will always be there for me.

I would like to express my greatest gratitude to my husband and lifelong collaborator Suat Tüçer for his endless love and support. He was the joy of these 6 years who made it easier and possible to finish my PhD. He always pushed me when I was stuck, and he did not let me drown. Now with his help, I am finishing this PhD thesis. It was a real luck that we worked in the same lab shoulder to shoulder, sometimes fighting sometimes cuddling. I thank you for enduring all my tantrums during my stressful times and making me laugh every time I cry.

Lastly, I owe my most special thanks to my dearest princess, my daughter Göksu. You are the everything of my life. Even though you were born in a challenging time, I am very grateful for having you. You are the main reason that I keep fighting and getting up every time. Thank you for being my daughter and for your unconditional love. I love you!

I thank Bilkent University UNAM for the fellowship that supported me during my PhD. This study was supported by TUBITAK 1001 program (TUBITAK, Grant No: 118S398).

TABLE OF CONTENTS

CHAPTER 1	1
Introduction	1
1.1. Characteristic Properties of Nanoparticles.....	2
1.2. Classification of Nanoparticles.....	3
1.2.1. Gold Nanoparticles	3
1.2.2. Silver Nanoparticles	3
1.2.3. Titanium Nanoparticles.....	4
1.2.4. Quantum Dots	4
1.3. Characterization of Nanoparticles.....	5
1.4. Applications of Nanoparticles	7
1.4.1. Medical and Drug Delivery Applications.....	7
1.4.2. Manufacturing Applications	9
1.5. Toxicity of Nanoparticles.....	10
1.5.1. Exposure Routes and Cellular Uptake	11
1.6. Impacts of Physicochemical Properties of Nanoparticles on Cellular Uptake.	14
1.6.1. Size	15
1.6.2. Shape	15
1.6.3. Surface Charge	16
1.6.4. Hydrophobicity	16
1.6.5. Surface Functionalization.....	16
1.6.6. Mechanisms of Toxicity.....	17
1.6.7. ROS Generation-Oxidative Stress.....	18
1.6.8. Genotoxicity	19
1.6.9. Mitochondrial Damage	19
1.6.10. Inflammation.....	19
1.6.11. Apoptosis.....	20

CHAPTER 2	21
3D Cell Culture and Sample Preparation for RNA-seq.....	21
2.1. Objective of the Study	22
2.2. Introduction	22
2.3. Materials and Methods	25
2.3.1. Media and Buffers	25
2.3.2. HepaRG Cell Culture	25
2.3.3. Characterization of Nanoparticles.....	25
2.3.4. Buffer Exchange of Nanoparticles	26
2.3.5. Viability Assay	26
2.3.6. 3D Spheroid formation and NP Exposure of HepaRG Cells.....	27
2.3.7. mRNA Isolation from HepaRG Spheroids	27
2.4. Results	28
2.4.1. Microscopy Images of Spheroids.....	28
2.4.2. Characterization of Nanoparticles.....	29
2.4.3. MTT Assay Results.....	31
2.5. Discussion	33
2.6. Conclusion.....	34
CHAPTER 3	35
Transcriptomic Analysis of Nanoparticle Toxicity	35
3.1. Objective of The Study	36
3.2. Introduction	36
3.3. Materials and Methods	41
3.3.1. Software and Packages	41
3.3.2. Quality Check of Raw Data with FASTQC	42
3.3.3. Read alignment with HISAT.....	42
3.3.4. Transcript assembly and quantification with StringTie.....	43
3.3.5. Differential expression analysis with Ballgown	44
3.3.6. Statistical Analysis.....	47

3.3.7. Gene Ontology/Hallmark and KEGG Pathway Analysis	48
3.3.8. Plots	48
3.4. Results	48
3.4.1. Quality Check and Alignment Scores	48
3.4.2. Gene Expression Profiles of Nanoparticle Treated Spheroids	50
3.4.1.1. Gene Ontology Analysis	53
3.4.1.2. Hallmark Gene Set Analysis	58
3.4.1.3. KEGG Pathway Analysis	60
3.4.2.1. Liver Specific Genes	62
3.4.2.2. Metal Ion Response Genes	64
3.4.2.3. Oxidative Stress Response Genes	66
3.4.2.4. Apoptotic Genes	67
3.4.2.5. Metastasis and Tumorigenic Genes	71
3.5. Discussion	73
3.6. Conclusion	78
CHAPTER 4	79
CONCLUSION AND FUTURE PERSPECTIVES	79
BIBLIOGRAPHY	81
APPENDIX A	90
APPENDIX B	204
FastQC Results	204
APPENDIX C	206
List of Top DEGs for each NP	206

LIST OF FIGURES

Figure 1: Smart nanoparticles are made of nanocarriers.....	8
Figure 2: Medical applications of nanoparticles.	9
Figure 3: Mechanism of phagocytosis of NPs for the cellular uptake of the cells. Nanoparticles are adsorbed by opsonins and by complement activation they are up taken inside the cell. Created with BioRender.com.....	13
Figure 4: Physicochemical properties of nanoparticles. [51].....	14
Figure 5: Toxicity mechanism of nanoparticles. Reprinted with permission from the reference [60]	18
Figure 6: Visualization and viability assessment of 3D spheroids of HepaRG cells. A) bright-field microscopy image of HepaRG spheroids. B) The viability of the spheroid is tested with Calcein-AM and PI. Then, another spheroid was fixed with 70% ethanol for 10 min and staining was done with Calcein-AM and PI as control for the viability.	28
Figure 7: MTT assay for varying concentrations of water in the cell culture medium....	29
Figure 8: Characterization of nanoparticles with TEM. The TEM images show the distribution and sizes of A) 3nmAuNP B) 15nmAuNP C) 80nmAuNP D) 12nmAgNP E) 40nmAgNP F) 80nmAgNP G) 25nmTiO ₂ NP H) 100nmTiO ₂ NP I) 300nmTiO ₂ NP J) CdZnSe/CdZnS/ZnS and K) InP/ZnS	30
Figure 9: Assessing the viability of the NPs at different concentrations on HepaRG cells. A) Viability of HepaRG cells that are treated with A) gold nanoparticles B) silver nanoparticles C) titanium nanoparticles D) quantum dots.....	32

Figure 10: Schematic representation of high-throughput RNA sequencing workflow. Reprinted from (Thomas et al., 2022) [82]	38
Figure 11: Representation of the RNA-seq analysis methodology. Reprinted with permission from the reference [87]	40
Figure 12: A sample txt file that will be used for the merge step of the StringTie.....	44
Figure 13: A sample phenotype_data file. Columns should be separated with a comma.	45
Figure 14: Gene expression profiles of nanoparticle treated HepaRG spheroids. Upregulated DEGs (red), downregulated DEGs (blue) and non-significant genes (grey).	51
Figure 15: Venn diagrams of the DEGs for each size and material of the NPs. Common genes between different NPs are also shown.	52
Figure 16: Gene ontology analysis of the DEGs of AuNP treated cells.	54
Figure 17: Gene ontology analysis of DEGs of AgNP treated cells.	55
Figure 18: Gene ontology analysis of DEGs of TiO ₂ NP treated cells.	56
Figure 19: Gene ontology analysis of DEGs of QD treated cells.	57
Figure 20: Hallmark gene set analysis of the DEGs.	59
Figure 21: Heatmaps of the CYP enzymes in the NP treated cells.	63
Figure 22: Heatmaps of the metal ion responsive genes.	65
Figure 23: Heatmaps of the oxidative stress responsive genes.	66
Figure 24: Heatmap of the apoptotic genes for gold nanoparticles.	67
Figure 25: Heatmap of the apoptotic genes for silver nanoparticles.....	68
Figure 26: Heatmap of the apoptotic genes for titanium nanoparticles.	69
Figure 27: Heatmap of the apoptotic genes for quantum nanoparticles.....	70

Figure 28: Heatmaps of the metastasis genes in NP treated cells.	71
Figure 29: Heatmap of the genes related to liver cancer.....	72



LIST OF TABLES

Table 1: Summary of the quality check of the RNA-seq raw data. 49

Table 2: Common genes among all 5 NPs are listed with FC and their cellular function.
.....53

Table 3: KEGG pathways that are enriched in DEGs for each NP.....60



CHAPTER 1

INTRODUCTION

Nanotechnology has gained the attention since the last century. The skill to be able to manipulate and regulate the interaction of atoms and molecules is a fundamental aspect of nanoscience. The origin of the word came from Greek word “nanos”, which can be translated as “petite”, where it can be related to nanometer size particles diffused in solid, liquid or gaseous phase. [1] Nanoparticles are the most well-known product of nanoscience. These materials have at least one dimension smaller than 100nm. [2] [3] They can be made of different materials such as carbon, metal, or organic molecules and they might have different shapes like spherical or cylindrical. [4], [5] The development interest in nanomaterials has increased because they offer unique or improved physical and chemical properties when comparing to their bulk counterparts.[6] Further, their numerous benefits—such as their framework advantages, size, unique physical and chemical features, extremely reactive surfaces and flexibility for alteration make them ideal. [7]

Nanoparticles are materials which have at least one of its dimensions ranging from 1nm to 100nm. They can be synthesized with different materials such as metals, semiconductors, ceramics and polymers. The optical properties of nanoparticles are

different than their bulk form because of their nano-sizes and high surface area. They are being investigated in terms of their unique properties to be used for different purposes. Nowadays thanks to their unique properties, in many fields' nanoparticle usage can be found which some examples are medicine, catalysis, liquid treatment, agriculture and so. Their surface properties and their quantum properties cause them to exhibit significantly different behavior when compared to them with a bigger dimension. All together those specificities lead nanoparticles to display improved or unique electronical, thermal, catalytic, magnetic, optic, mechanic specialty. [6]

1.1. Characteristic Properties of Nanoparticles

Attracting properties of NPs increased their usage in diverse fields such as medicine/ drugs, food, cosmetics and industry. [8], [9] Their unique characteristics come from their large surface area, mechanical properties, optical and chemical reactivity. [10] They also have the ability to absorb and reflect visible and near-infrared light. There are two different factors that enable nanoparticles to have unique properties against the bulk materials of them which are surface effects and quantum effects. They have surface effects due to their large surface area, more fraction of atoms situated at the surface of the NPs and also surface atoms have less direct neighbors. [11] That is why NPs have different properties than their bulk counterparts. NPs also have properties that are affected by quantum phenomena which is because of the quantum confinement of the charge carriers in the physical dimensions. Some non-magnetic bulk materials exhibit magnetic properties in nano scale because of the quantum confinement. [6]

1.2. Classification of Nanoparticles

1.2.1. Gold Nanoparticles

Gold nanoparticles (AuNPs) are nanoparticles made of gold ions. They have unique optical, electronic, and chemical properties. These properties enable gold nanoparticles to have applications in diverse fields such as drug delivery, cancer therapy, diagnostics, biosensing etc. [12] They are highly biocompatible and they are easy to be functionalized with biomolecules or drugs. Their controlled size, shape and surface properties offer exceptional discoveries. They are also utilized in electronic chips. Their electrical conductivity and tunable plasmonic behavior enables developments in the performance of the electronic devices. [13]

1.2.2. Silver Nanoparticles

Silver nanoparticles, which is another important metal nanoparticle can be found in many applications which some of them include food industry, cosmetics, health care, etc. Those nanoparticles own this diverse usage area by their distinctive chemical and physical features. Those features include antibacterial, antiviral and antifungal effects. [14] Silver, known with its antimicrobial properties, is used in house cleanings, products used in health care like wound dressing, inside cosmetics or may take part in the production of medical devices and so. Nanotechnology uses its absorption and scattering properties for the advancement of new products. [15] Silver nanoparticles started to be used in biomedicine as nanotechnology advanced. [16] They are used in biological sensing area as optical and spectroscopic tags. [17]

1.2.3. Titanium Nanoparticles

From the time of the first commercial production in the early 20th century, titanium dioxide (TiO₂) has had a place in varied fields. Some of the fields that TiO₂ took place include sunscreens or as pigments like in paints, in toothpaste and many more can be described. The identification of TiO₂ electrode to phototactically split water under ultraviolet light, let TiO₂ to be further investigated in different areas which can be separated as environment and energy. [18] Currently they are employed in photodynamic therapy, leveraging their effective photo oxidation capabilities. When exposed to TiO₂ nanoparticles that had been photo-excited human colon carcinoma cells could be eliminated. [19] In addition, 20 nm size TiO₂ particles when irradiated can also be efficient for eliminating cells of pulmonary cancer. [20] Another example of improved functionality can be found in sunscreen products as UV filters. [21] TiO₂ is being used in cancer treatment. TiO₂ NPs are combined with targeted drugs for the targeted delivery of the drug molecules, they also induce DNA damage and activate apoptotic pathways. [22]

1.2.4. Quantum Dots

Quantum dots (QDs) have unique physical and optical characteristics. They have the capability of binding biomolecules which makes it a good candidate for being a biosensor and biological imaging agent. [23] Targeted drug delivery enables specific targeting the drug molecules into tumor sites decreasing the probability of the drug going into the healthy cells. [24] Magnetic NPs in drug delivery have several advantages. They can be easily visualized. Also, they can be heated in magnetic field to induce on-time drug delivery. [25] Lipid nanoparticle (LNP) technology is rapidly gaining interest in the fields

of oncology and infectious disease vaccination, gene editing techniques, and emerging mRNA and replicon-based therapies. [26]

1.3. Characterization of Nanoparticles

Materials at the nanoscale often exhibit distinct features compared to their bulk forms because their high surface-to-volume ratio significantly boosts molecular reactivity. These unique features of nanoparticles (NPs) can vary greatly that can include differences in optic, chemical behaviors, mechanic, and electronic properties. Because of the academic significance and potential technological utilization across various fields, make them the subject of extensive research. Today, a great variety of nanomaterials are being produced compared to just 10 years before with a larger quantity. High accuracy and reliable methods for characterizing those nanoparticles became a necessity. [27]

X-Ray Photoelectron Spectroscopy characterizes nanoparticles considering that the depth of the surface layer involved in initial interactions alongside the surrounding is only some nanometers, instrument that can analyze nanoscale surface level are crucial. XPS has also the ability to provide data about the elements identities which make it an exceptional instrument. [28]

To analyze crystalline materials, X-ray diffraction (XRD) is a highly effective non-damaging technique. It allows for the evaluation of various structural aspects, including the identification of favored crystal alignment, frameworks and phases, and additional information like typical grain dimension, imperfections of the crystals and crystal quality. The intensities observed in the X-ray diffraction (XRD) analysis indicate if the produced nanoparticles are inherently in crystal or amorphous structure. [29]

The atomic force microscope (AFM) is one of the most valuable device due to its capacity to give information about structure of the nanoparticle investigated, doing this in aqueous or in air environment. [30] Using the Atomic Force Microscope (AFM), single nanoparticles or many of them can be found with high precision. Unlike other microscopy approaches, the AFM provides detailed depiction and analysis in 3D. [31]

The Fourier Transform Infrared (FTIR) technique is one kind of spectroscopy used to analyze variation in whole content of biomolecules by detecting variations in functional groups. The Fourier Transform Infrared analyzes how infrared radiation affects the vibration and rotation of molecules at specific wavelengths. By observing these effects, FTIR can identify configured variations and molecular binding changes, providing insights into relationships between different entities. [32]

Dynamic light scattering is a valuable technique used across many scientific fields. This method allows researchers to study specific properties of molecules by analyzing how they scatter light. In a usual light-scattering study, a specimen is exposed to a monochromatic light wave, and the scattered light is detected by a relevant detector. [33]

Nanoparticles exhibit optical features which can be highly responsive to variations in dimension, structure, agglomeration, and density. The distinctive optical behavior of metal nanoparticles arises from the combined oscillations of free electrons. If these electrons are stimulated by electromagnetic radiation, they produce phenomena known as surface plasmon polariton resonances. These alterations affect the refractive index near the outer layer of the nanoparticle which UV–VIS spectroscopy can be used to analyze

nanomaterials effectively. This method is quick and straightforward to use, making it particularly suitable for analyzing nanoparticles, in particular those in emulsions. [34]

1.4. Applications of Nanoparticles

With the enormous development in nanotechnology, nanoparticles are being used in various fields, including medicine, industry, energy, cosmetics and material science. Changing properties of nanoparticles as we scale down to nanometer, enables us to use the nanoparticles in different purposes. They can be used as environmental remediation to remove the pollutants from the soil and water, as food additives to increase the flavor and stability, as drug delivery agents which can be targeted directly to the target site in the body, which increases the drug stability and decrease side effects.

1.4.1. Medical and Drug Delivery Applications

In cancer treatment, drug delivery to the tumor site is important, because other treatments such as chemotherapy are not selective to the tumor site, and they damage all the cells in the body starting from rapidly dividing cells such as hair. We need for specific and selective treatments for the diseases, so that the treatments will be more effective and less harmful to the body. Nanoparticles are being used in cancer treatment for the smart drug delivery purposes. Smart nanoparticles that are modified and targeted to the tumor site. They can deliver drug in a more efficient and targeted way for example water-insoluble drug can be delivered by using micelles. [35] There are many kinds of nanocarriers that can be used for formation of smart drugs such as polymer based nanocarriers, biomimetic-based smart nanocarriers etc. (Figure 1.1)

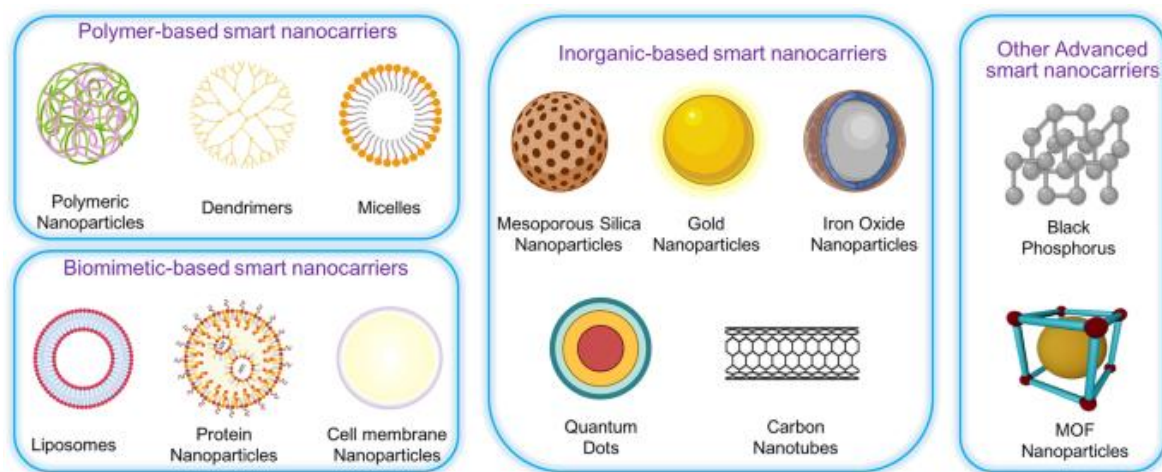


Figure 1: Smart nanoparticles are made of nanocarriers. Reprinted with permission from ref. [35]. Copyright 2023 SpringerNature

Drugs that are carried with nanoparticles are less detrimental to the human health above all more efficient since the durability of the drugs are protected by the nanoparticles. [36] Doxorubicin that is a chemotherapy drug is found to be less toxic when they are delivered with liposomal structure compared to the free doxorubicin in metastatic breast cancer cells. [37] Drug resistance is another problem in cancer treatment, so overcoming the drug resistance in cancer is big step in terms of the treatment. Nanoparticle aided drug delivery can help overcoming the multiple drug resistance in cancer treatment. [38] Smart drug delivery systems are being set for stimulus responsive nanoparticles which will carry the drug to the tumor microenvironment and release its cargo when it is stimulated by a molecule at the site. [39] Nanoparticles has many advantages compared to conventional treatments including targeted delivery of the drug, more durable drug, and less side effects. They should be investigated more for finding new strategies to improve treatments for every disease, because engineered nanoparticles might be used in various purposes. (Figure 1.2)

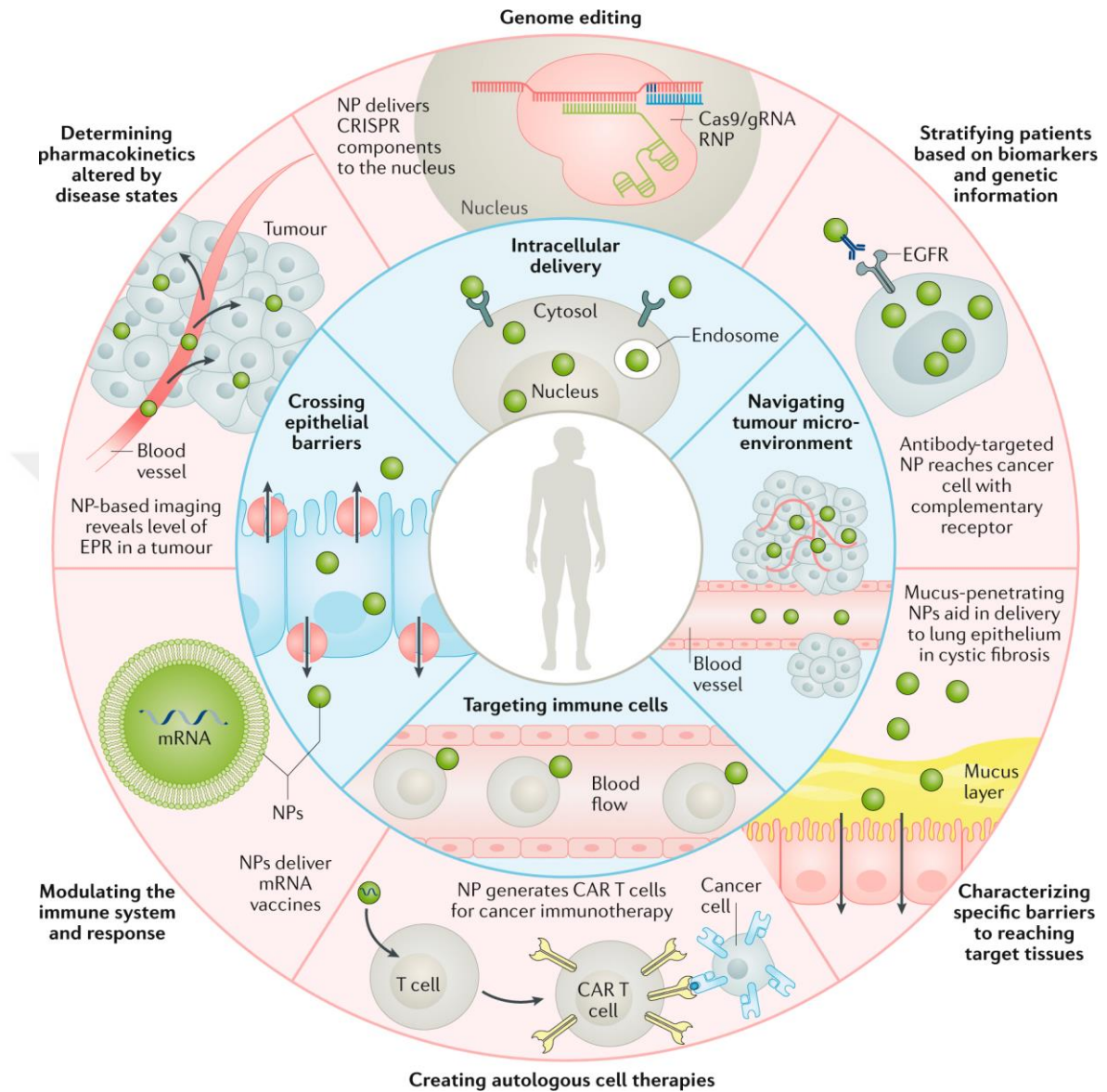


Figure 2: Medical applications of nanoparticles. Reprinted with permission from ref. [40]

1.4.2. Manufacturing Applications

Nanotechnology produces new materials which have unique properties. These materials help to develop new techniques in manufacturing. Nanoparticles are already being used in various products including cosmetics, paints, textiles and electronic devices. In the

cosmetics field nanoparticles are used for the long-lasting, antibacterial and antifungal effects, increased stability, as well as the for the UV-filtering purposes in sunscreens. [41] Paints were enhanced by nanomaterials in terms of their stability, color, fire retardant, thermal insulation, UV-protection, water repellent, antibacterial and self-cleaning properties. [42] Also, ceramic nanoparticles provide hard, wear resistant, temperature resistant, self-cleaning and antibacterial products. In textile industry, nanoparticles are used to produce smart fabrics that self-clean, prevent UV-rays, prevent electrostatic effects, repel water, prevent radiation etc. [43] They have antibacterial properties which makes them used as conservatives in cosmetics and food products.

To find the the potential risk of the use of nanoparticles in the food products, an extensive literature search and inventory by the European Food Safety Authority (EFSA) resulted in finding of 779 sources related to the toxicity of the nanoparticles in the food products. [44] All these references shows the potential negative impacts of NPs on human health and environment. Also, the Food and Drug Administration (FDA) has some criteria regarding the regulation of NPs. Two factors are considered when determining if FDA-regulated products incorporate nanotechnology. Size of the particles and nanoscale related characteristic properties are the two factors that are considered. When considering the safety of a product, both the industry and the FDA must evaluate these factors. [45]

1.5. Toxicity of Nanoparticles

Nanoparticle cytotoxicity is a recent concern that has arisen after the diverse usage of them in various products and purposes. They are being used in many areas such as cosmetics, industry, medicine and food products so they are in our daily lives. This diverse

use of NPs necessitates the need for the investigation of those particles in terms of their effects on human health. Nowadays, there are many studies on biosafety and cytotoxicity of the NPs. The cytotoxicity of NPs starts with their entrance to the cells. They are generally taken up by the cells by energy-dependent pathways as well as they could also enter the cells with energy-independent pathways such as diffusion. Energy dependent pathway is endocytosis.

1.5.1. Exposure Routes and Cellular Uptake

Human exposure to nanoparticles occurs via ingestion, respiration, skin or intravenous administration. According to types of route of exposure it was found that the intravenous administration of the nanoparticles resulted in 100% bioavailability whereas other route of exposures had different percentages of bioavailability depended on the properties of the nanoparticles. [46] Nanoparticles spread to all the body including brain and brain accumulation of nanoparticles causes disruption of the blood brain barrier and hippocampal stem cell quiescence. [47] [48] When nanoparticles are ingested they are not fully absorbed in the gut.

Human body is exposed to nanoparticles with the products that is used every day. For example, beer may contain clay nanoparticles, sunscreens contain titanium dioxide nanoparticles, bicycles may contain carbon nanoparticles, and clothing product may contain silver NPs. (Love et al. 2012) In addition to products that contain synthesized NPs, human body is exposed to nanoparticles that naturally occurs in the environment by processes such as combustion, atmospheric aerosols, and volcanic activity. [2] However, the impact of naturally occurring NPs is related to the level of their accumulation in living

organisms and their damage to these organisms. Produced nanoparticles are modified by surfactants and stabilizers which enables them to be durable and resist against aggregation. Therefore, it is essential to assess the potential impact of the NPs on human body. [49]

Cellular uptake of the NPs changes their fates in the cell. Polar or charged molecules cannot get inside the cell by the hydrophobic membrane. These molecules have to be internalized by the process called endocytosis. Endocytosis is the main entrance route of the NPs to the cells. This process is energy-dependent, and it has two categories: phagocytosis and pinocytosis.

Phagocytosis is the energy dependent cellular uptake of particulate matter (0.5-10 μm) via plasma membrane envelop. Cells engulf cells or other particles in this process. For immunologic defense mechanism, this process is very important because cells that does the phagocytosis clears the infections, dead cells and cell debris, and they kill the pathogens. [50] NPs are phagocytosed by the adsorption of opsonins on the NPs, they interaction of the opsonins with related receptors which enable the internalization of the NPs. (Figure 1.3) Inside the cell a phagosome is formed simultaneously with phagocytosis. At later stages, this phagosome is fused with lysosome and the particles are digested with the acidic pH and enzymes of the lysosome.

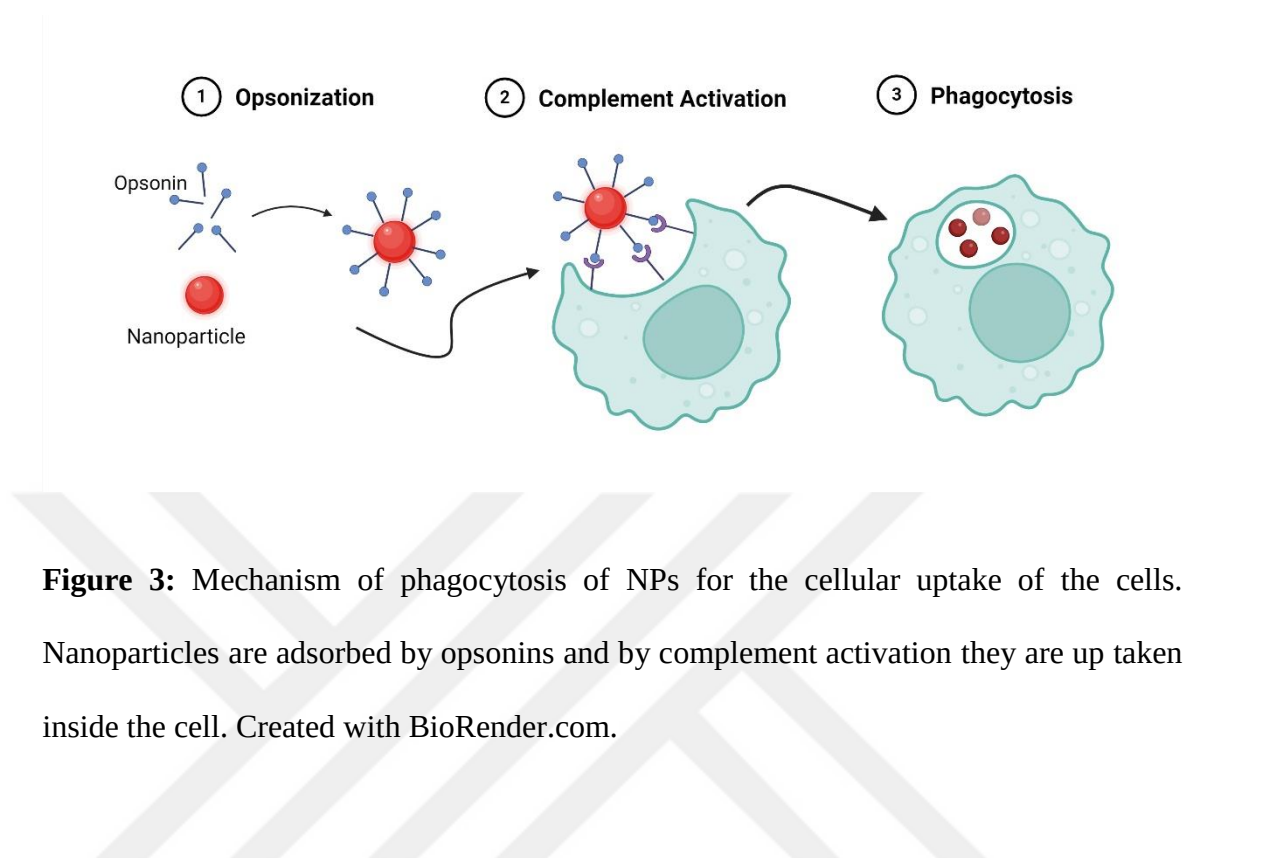


Figure 3: Mechanism of phagocytosis of NPs for the cellular uptake of the cells. Nanoparticles are adsorbed by opsonins and by complement activation they are up taken inside the cell. Created with BioRender.com.

Pinocytosis is the cellular uptake of the fluids and dissolved solutes from the extracellular environment. It is usually used for the internalization of nanometer sized small particles. [51] It is also called cellular drinking which involves up taking a small droplet of extracellular fluid. At the pinocytosis process, a pinosome is formed and it later fuses with lysosome to hydrolyze and break down the content. Pinocytosis has 4 subcategories which are clathrin-mediated endocytosis, caveolae-mediated endocytosis, clathrin and caveolae independent endocytosis and micropinocytosis. In clathrin-mediated endocytosis specific particles in the extracellular matrix bind to the receptors on the cell membrane. By this complex, clathrin-coated vesicle is formed inside the cell. These vesicles eventually fuse with lysosomes via early endosomes. Caveolae-mediated endocytosis involves the formation of bulb-shaped membrane invaginations which are called caveolae. This process is involved in cell signaling and regulation of membrane components. [52]

Nanoparticles are taken inside the cells by one of these processes. The process that is used for uptake of each type of the nanoparticle is chosen according to the physicochemical properties of each NP.

1.6. Impacts of Physicochemical Properties of Nanoparticles on Cellular Uptake

Nanoparticles are mostly taken up by the cells by endocytosis. The mechanism and type of endocytosis depends on the physicochemical properties of nanoparticles. The factors that affect the internalization of the NPs are size, shape, surface charge, hydrophobicity and surface functionalization. (Figure 1.4)

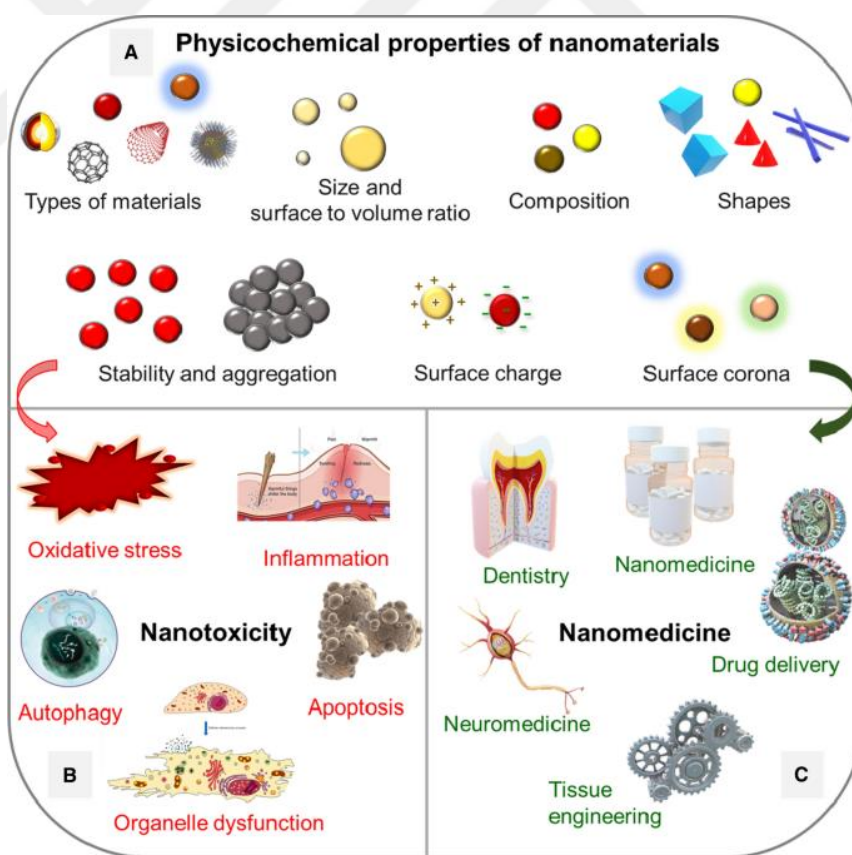


Figure 4: Physicochemical properties of nanoparticles. Reprinted with permission from ref. [53]

1.6.1. Size

Size of the nanoparticle is the main factor in choosing the uptake pathway as well as the efficiency of the uptake. Nanoparticle dimension is crucial in defining the features and effectiveness of materials used in medical uses. Numerous specialized nanomaterials and biofunctionalized nanoparticles are created to control biochemical reactivity, regulate biological activity, and for medicinal purposes. In medicinal purposes the NPs dimensions are very crucial. Inorganic nanoparticles are generally influenced by their dimensions because of their chemical and physical features. To be able to use their features controlling their size is necessary. Also, the hydrosol features of nanoparticles necessitate that their dimensions be kept below 100–200 nm, because bigger ones tend to aggregate due to gravity-driven forces. Biological applications generally necessitate specific dimension limitations because of factors such as spatial interference in the process of binding, constraints imposed by particular physiological activity, or dimension-based preference by particular biological sections. [54]

1.6.2. Shape

Biological entities exhibit a wide variety of shapes, like cylindrical tobacco mosaic viruses, round HIV, multi-form cells in the human body, and spiky form bacteria. This diversity in shape provides evolutionary benefits tailored to particular purposes, including surface interactions, simple transport, and active transport. [55]

1.6.3. Surface Charge

The surface charge of nanoparticles (NPs) plays a critical role in their systemic localization, correlation with physiological surroundings, and cell penetration. Typically, positively polarized nanoparticles have the ability to better penetrate inside cells compared to balanced or negatively polarized nanoparticles, because of the negatively polarized inherent feature of cellular membranes. [56]

1.6.4. Hydrophobicity

A crucial chemical-physical feature of nanoparticles (NPs) is hydrophobicity, which considerably influences their relations and the outcome in biological surroundings. At the time of reacting in living systems, hydrophobicity of nanoparticles has got a crucial impact in shaping living systems outcome and toxicity that they may cause. This consist of the outer layer formation made of protein and lipid, bio-accessibility, immunological reactions, gathering in the body, and internalization inside cell. [57]

1.6.5. Surface Functionalization

Applying nanoparticles in healthcare demands precise monitoring over their relations to biological molecules. For instance, nanoparticle outer layer structures can be engineered to prevent cytotoxic effects, increase their uptake inside the cell, and increase the binding ability of their content, which are crucial for better internalization inside cell. Additionally, the designed functional outer layer on the nanoparticles has the possibility to be adjusted to achieve targeted or distinct recognition for biosensor technology applications. Even the ability to avoid relation, or being immune-inert has the potential to

be crucial, as these features can help to stop particles from being detected and caught by the body's defense mechanism. [58]

1.6.6. Mechanisms of Toxicity

Nanoparticle toxicity depends on the route of exposure, surface charge, size and chemical composition of the nanoparticle. These properties of nanoparticles induce reactive oxygen species (ROS) production. ROS generation in the cells cause oxidative stress, inflammation, DNA damage, ceasing cell cycle and finally cell death.

Although NPs have various advantages, there are also limitations of using them. Small size of them leads high surface-to-volume ratio of NPs. High surface area of NPs causes them to expose stress on the cells which results in changes in gene expression profiles of the cells, DNA damage, production of reactive oxygen species (ROS), lipid peroxidation, mitochondrial dysfunction and DNA damage. (Figure 1.5) These impacts on the cells causes several health problems in the human body. [59] Since the human exposure to these NPs continually increasing, their impact on the human health needs detailed investigation. To assess the potential risk of damage to individuals exposed to nanoparticles, we need to explore the toxicity mechanism of nanoparticles in the cells. [60] Many studies showed that NPs induce pro-inflammatory responses in the cell. Nanotoxicology has found that nanoparticles cause more severe damage on the cells causing inflammation, DNA damage, ROS production compared to equal amount of the same material and substance with larger size. [61]

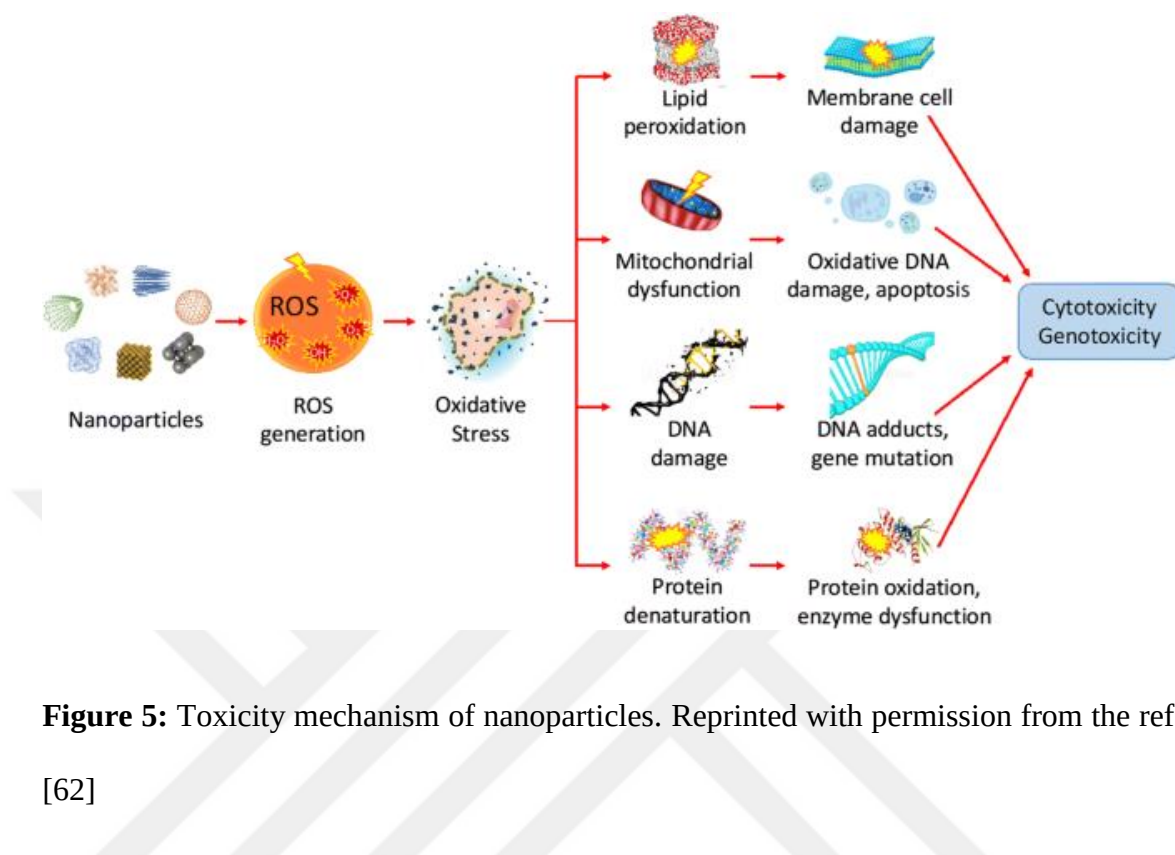


Figure 5: Toxicity mechanism of nanoparticles. Reprinted with permission from the ref. [62]

1.6.7. ROS Generation-Oxidative Stress

Oxidative stress is a key factor in the toxicity of nanoparticles. Different sort of nanoparticles can trigger oxidative stress with the production of reactive oxygen species (ROS) within cells. [63] Non cellular aspect like the nanoparticle outer layer, contents, metal constituents, and dimensions are all a possible reason for oxidative stress. [64] Key contributors to ROS formation include the internalization inside the cells of nanoparticles and the secretion of metal ions inside cells. Additionally, correlation between nanoparticles and cells may cause mitochondria deficiency, further promoting ROS production. In living organisms, nanoparticles can stimulate the release of cytokines, that later lead to secondary oxidative stress through the production of radical species and ROS. [65]

1.6.8. Genotoxicity

Cell functions like DNA replication throughout cell division, RNA synthesis, Protein synthesis, cell specialization, and cellular life stages are crucial for preserving the framework and effective integrity of genomic content (DNA and RNA). Because of their tiny size, nanoparticles near cells are more likely to be internalized. Once inside, these nanoparticles' reaction with DNA are increased, which may cause physical and activity-based alteration such as DNA malfunction/correction and methylation of the DNA. These changes can have detrimental consequences on the function of the cell and subsequently on the overall wellness. [66]

1.6.9. Mitochondrial Damage

Mitochondria can be one of the organelles that nanoparticles can interact with, which makes this situation an important reason for toxicity. Research has demonstrated that nanomaterials can lead to change in activity of the mitochondrial permeability transition pore (MPTP) and affect in metabolic efficiency, changes in structure, disruption of electrochemical gradient, and increase cytochrome C secretion. [67]

1.6.10. Inflammation

As soon as inside the living being, nanoparticles (NPs) interfere with regular biological functions and induce immune reaction. Additionally, nanoparticles have the ability to impact the body's defense mechanism by initiating distinct routes of action, which may operate individually or coordinate reciprocally. They may stimulate inflammasome signaling pathways. Besides nanoparticles might implicitly disrupt the balance of essential

constituents due to the secretion of ions, potentially affecting immune system activity. In addition, the immune system may either recognize nanoparticles (NPs) or overlook them but once detected, NPs can trigger inflammatory reactions, which latterly the immune system can mount an action to neutralize those NPs. Furthermore, beyond toxic ones, non-toxic NPs as well is believed to alter the typical protection mechanisms to toll-like receptor (TLR) ligands. [68]

1.6.11. Apoptosis

Apoptosis is a controlled process of cell disruption, whereas necrosis is an unregulated form of it. The two types of cell disruption may result from nanoparticles (NPs) interaction. Apoptotic cell disruption is generally not causing inflammation, whereas necrotic cell disruption may induce inflammation. These two routes represent limits, and various situations involve an intricate interplay of both processes. The cell disruption mechanisms are influenced by various factors, including the amount, interaction time, and the polarity of the outer layer of nanoparticles (NPs). [69]

CHAPTER 2

3D Cell Culture and Sample Preparation for RNA-seq

The work described in this section is partially published in the following study [70]:

Merve Erden Tüçer ^a, Nazlıcan Tunç ^a, Suat Tüçer ^a, Rana Acar ^b, Duygu Deniz Usta ^c,
Kouroush Salimi ^d, Özlen Konu ^{a b}, Urartu Özgür Şafak Şeker ^{a*}

a: UNAM–Institute of Materials Science and Nanotechnology, National
Nanotechnology Research Center, Bilkent University, 06800, Ankara, Turkey

b: Bilkent University, Department of Molecular Biology and Genetics, 06800, Ankara,
Türkiye

c: Gazi University, Faculty of Medicine, Department of Medical Biology and Genetics,
06500, Ankara, Türkiye

d: Ankara Yıldırım Beyazıt University, Faculty of Engineering and Natural Sciences,
Department of Chemical Engineering, 06010, Ankara, Türkiye

*Corresponding Author: Urartu Özgür Şafak Şeker, email: urartu@bilkent.edu.tr

Reprinted with permission of authors from following publications;

<https://doi.org/10.1016/j.cbi.2024.111303>

2.1. Objective of the Study

Nanoparticle usage in daily life and in medicine increases the exposure of the human body to these materials, and this exposure continues increasing every day. This high potential of human exposure raises the concerns about the biosafety of NPs. Nowadays, there is not a detailed study that works on the mechanism of toxicity by exploiting diverse nanoparticles with differing materials and sizes. In this study, we aim to characterize eleven different NPs and expose them to HepaRG cells, which is a good alternative to primary hepatocytes.

2.2. Introduction

The liver is the main organ for detoxification of xenobiotics and nanoparticles. For the improvement of our comprehension about the nanoparticle toxicity, we need more complex and detailed analysis of the toxicity mechanism. An extensive pathway analysis is essential to reveal the underlying mechanism of nanoparticle toxicity.

Although most hepatic cell lines are derived from tumors and have a very low capacity of drug metabolism, there are some cell lines that are good alternatives to primary hepatocytes. HepaRG cell line is the first cell line that have the ability of in vitro differentiation into mature hepatocytes. [71] HepaRG cells are considered as the best alternative to the primary hepatocytes for the studies including differentiation, liver metabolism, drug effect/metabolism/toxicity. Since they have the cytochrome enzyme levels comparable to the primary hepatocytes. [72] Tumor models in 2D cell cultures have altered expression profiles compared to 3D, lack of tumor heterogeneity and microenvironment. [73] The limitations in the 2D systems induced the development of the

3D cell culture system which involves in vivo-like cell-cell and cell-extracellular matrix (ECM) interactions that are necessary for the regulation of gene expression profiles and cellular function. [74] 3D culture of the cells provides improved cell-cell and cell-extracellular matrix interactions. Also, cells are able to respond to external stimuli which similar to in vivo. [75] For mimicking the cellular microenvironment, in vitro models are being improved for providing the 3D properties of the organs in the body. [76] All these reasons convinced us to use 3D cell culture for the HepaRG cells. Because we want our cells to have gene expression profiles that are comparable to in vivo, since the result of the project will be used for the determination of the toxicity of the NPs in vivo.

Cell culture is one of the most important techniques for molecular and genetic analysis in the field. 2D monolayer culture is one of the most common culturing method of the cells which requires the cells attach and grow on a flat surface. In 2D culture, the interactions between cells and cells-microenvironment get limited, this causes the cell to change their gene expression profiles and their metabolism. Also, the changes in the gene expression profiles and mechanism causes getting incorrect results about the toxicity mechanism of the NPs and responses of the cells against these toxic agents, since they are cultured in 2D. 3D cell culture provides cells cell-cell junctions and better communication through microenvironment. In 3D culture, since cells will have in-vivo like settings, they will respond to toxic molecules more comparable to in vivo. Thus, we decided to use 3D cell culture system for the HepaRG cells before exposing them with NPs for the transcriptomic analysis. We generated our HepaRG spheroids by using agarose coated well plates. In this cell culture system involves the cells to attach to each other rather than attaching to the flat surface in the well. Incubation of the cells on agarose coated plates for 4 days resulted

in the formation of spheroids. Viability assessment of the cells inside the HepaRG spheroids is tested with staining with Calcein-AM which stains the live cells. Then, the spheroids are stained with propidium iodide (PI) for visualizing the dead cells. Both staining showed that the cells inside the spheroid are alive.

RNA seq is a high-throughput sequencing technique that sequences whole RNA molecules from the cells. This system helps transcriptomic analysis to be more detailed, precise and more informative. Long non-coding RNA, miRNA, siRNA and mRNA are all sequenced by RNA-sequencing. [77] RNA-seq helps to better understand the gene expression changes, finding alternative splicing and finding novel transcripts with its deep profiling capacity. Many pathways and mechanisms could be investigated by the help of RNA-seq because it gives information about expression levels of each gene even though they are low abundance. Deep profiling of the transcriptome helps to find the responsible genes for the pathological and physiological disorders. RNA-seq analysis methodology involves sample preparation, sequencing and bioinformatic analysis of the data. [78]

In this study, we chose four different types of nanoparticles, which are gold, silver, titanium and quantum dots, together with HepaRG spheroids for establishing a system for evaluating toxicity of the nanoparticles. As a result of the RNA-seq analysis, we got detailed information about the responses of the HepaRG cells to the NPs toxicity. We found the genes that are significantly dysregulated upon NP exposure. Our methodology combines 3D cultures with nanoparticle toxicity to elucidate genes reacting to this toxicity.

2.3. Materials and Methods

2.3.1. Media and Buffers

HepaRG cells (Cat.No. HPRGC10) were purchased from Thermo Scientific. William's E Medium(WEM) (Cat.No: W1878) and hydrocortisone were purchased from Sigma Aldrich. Human insulin was purchased from pharmacy (Humulin N, 100U/mL). Full WEM media is prepared as 10% FBS, 100 U/ml Pen/Strep, 2mM L-glutamine, 32 mU/mL human insulin and 20 µg/mL hydrocortisone.

2.3.2. HepaRG Cell Culture

HepaRG cells were cultured in William's E Medium supplemented with 10% FBS, 100 U/ml Pen/Strep, 2 mM L-glutamine, 32 mU/mL human insulin and 20 µg/mL hydrocortisone. HepaRG cells were trypsinized with 2 mL trypsin for 2-3 min, then added with 8 mL full WEM, then they were centrifuged at 2,000 rpm for 5 min. After discarding the supernatant, the cell pellet was resuspended in 13mL full WEM and seeded to T75 flask. The cells were cultured until 80% confluence changing medium every 2-3 days.

2.3.3. Characterization of Nanoparticles

NPs are obtained from a collaborator and characterized with TEM. TEM grids were added 10 µL of each NP, and incubated 10 min. Then the NPs on the grid were discarded, and they were washed with distilled water 2 times. Then dried at room temperature for 10 min before analysis.

2.3.4. Buffer Exchange of Nanoparticles

Buffer exchange of the nanoparticles was done with 10 kDa ultra-filtration centrifugal tubes. 10mL of nanoparticle was added into the ultra-filtration centrifugal tubes and then they were centrifuged until 5mL of solution was left. Then, 5mL of full WEM was added drop by drop by swirling the tube. Then, it was centrifuged again until 5mL of solution was left. The centrifuge and media addition step were repeated at least 4 times, so that the volume of the solution that the nanoparticles were synthesized is reduced to less than 1 mL in the total volume.

2.3.5. Viability Assay

T25 flasks were used for the culturing of HepaRG cells until 90% confluency. Then the cells are collected with trypsin, then they seeded into the 96 well plates in a density of 30,000 cells/well and left for attachment for 1 day. Later, the cells are treated with varying concentrations of nanoparticles in fresh medium. After incubating for 24hr, medium was discarded and new media is added containing 10% MTT. After 4h, medium was discarded, 100 μ L DMSO was added on the cells to dissolved the formazan crystals. Absorbance values were measured with microplate reader at 560nm. According to the results, we used 20 μ g/mL, 10 μ g/mL, 10 μ g/mL for 3nm, 15nm and 80nm AuNP respectively. We used 20 μ g/mL, 50 μ g/mL, and 200 μ g/mL nanoparticles for 12nm, 40nm and 80nm AgNP, respectively. We used 20 μ g/mL, 10 μ g/mL, 10 μ g/mL for TiO₂NPs. And we used 20 μ g/mL and 60 μ g/mL for the QDs.

2.3.6. 3D Spheroid formation and NP Exposure of HepaRG Cells

1.5% cell culture grade agarose was dissolved in sterile PBS and autoclaved. Then 50 μ L of agarose was added into each well of 96 well plate and 2mL of agarose was added into the each well of 6 well plates. Then, HepaRG cells were seeded into the plate in a density of 3,000 cells/well in 100 μ L medium. Cells were cultured for 4 days before they are collected as spheroids. On day 4, the spheroids were collected into one well of agarose coated 6 well plate. Then, spheroids were incubated with the certain amount of nanoparticles for 24 hr. Next day, the spheroids were collected by centrifugation and total mRNA isolation (M&N, NucleoSpin RNA, Cat.No.740955.50) was done according to the manufacturer's protocol.

2.3.7. mRNA Isolation from HepaRG Spheroids

Total RNAs that are isolated from HepaRG spheroids (treated with gold nanoparticles (3nm, 15nm, 80nm), silver nanoparticles (12nm, 40nm, 80nm), titanium nanoparticles (25nm, 100nm, 300nm) and 2 quantum dots for 24 hours), (each nanoparticle and control group has 2 replicates). Total RNA was extracted using NucleoSpin (Macherey-Nagel™ NucleoSpin™ RNA, Mini Kit) following the manufacturer's protocol. The quantity and quality of RNA were examined by Thermo Scientific™ NanoDrop™ 8000 Spectrophotometer. Library sequencing was performed on an Illumina HiSeq.

2.4. Results

2.4.1. Microscopy Images of Spheroids

3D HepaRG Spheroids were formed on agarose-coated wells, then they were transferred to microscopy slides for visualization. Then, one spheroid was stained with Calcein-AM and propidium iodide (PI) dyes and visualized under related filter in inverted light microscopy. Then, one spheroid was fixed with 70% ethanol for 10 min, and it was stained in the same way with the other spheroid with Calcein-AM and PI for 5 min. (Figure 6)

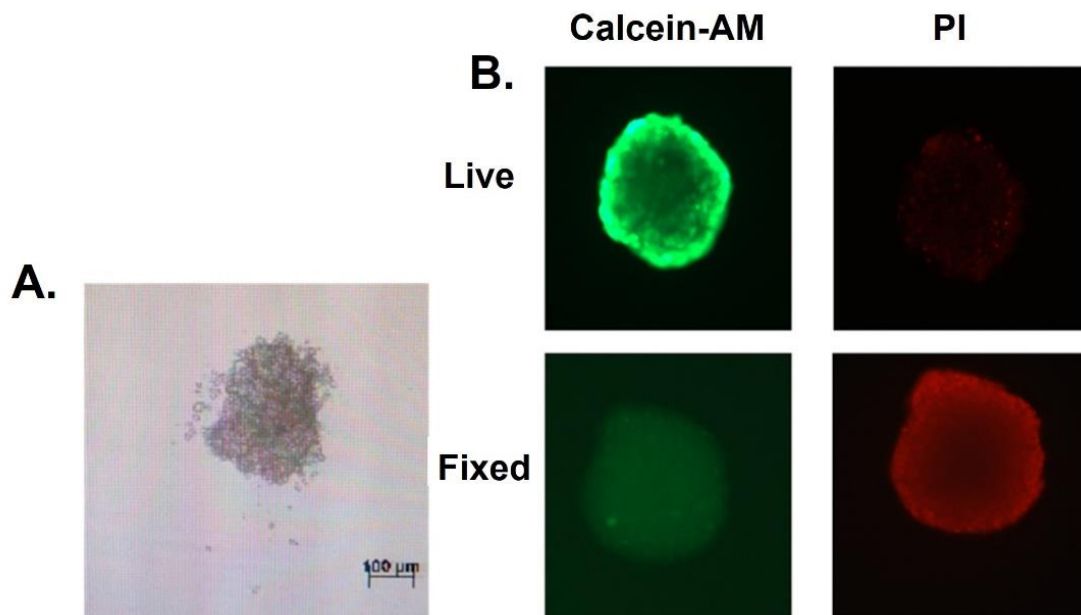


Figure 6: Cell viability evaluation of HepaRG spheroids. **A)** Bright-field image of the spheroid. **B)** Viability of the cells inside the spheroid is tested. Calcein-AM stains viable cells, and PI stains dead cells. Then, a spheroid was fixed with 70% ethanol for 10 min, then stained with the same dyes for the control group.

2.4.2. Characterization of Nanoparticles

NPs were exchanged into water at the beginning. However, water exposed additional toxicity on HepaRG cells as we tested it with MTT viability assay with varying water concentrations in cell culture medium. As seen in Figure 7, even small percentage of water in the cell culture medium decreased cell viability.

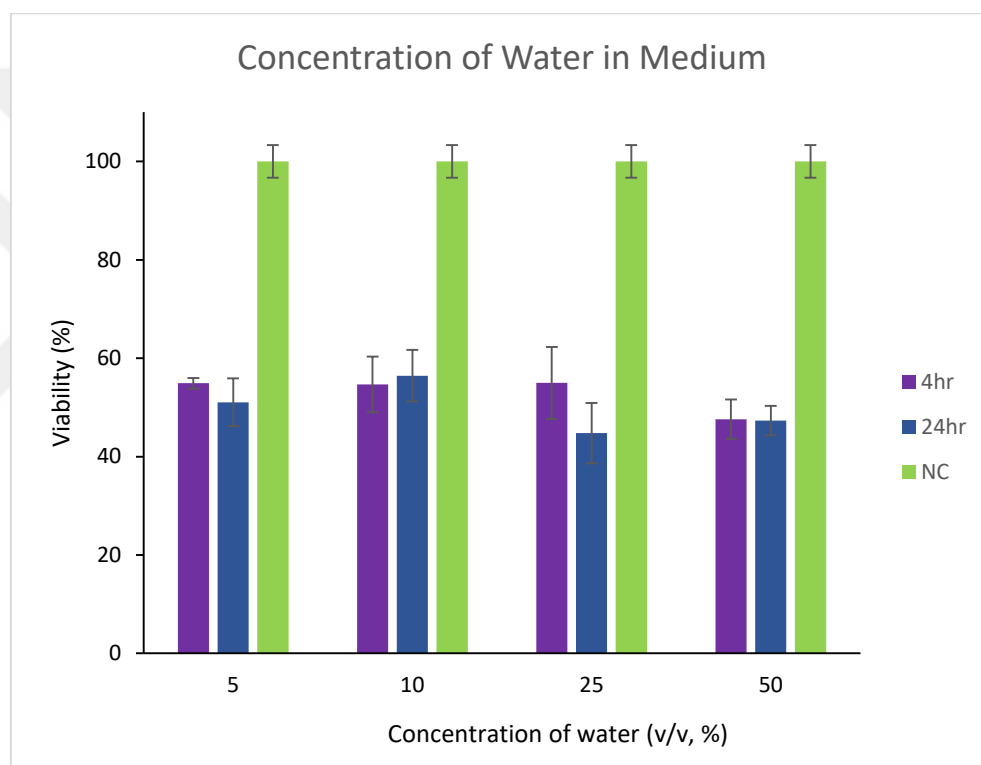


Figure 7: MTT assay for varying concentrations of water in the cell culture medium.

Characterization of NPs with TEM is important because this technique allows the size measurement of NPs and see the shape and distribution of them in the cell media. After NPs are exchanged into full media, the nanoparticles are visualized under TEM microscopy. All the nanoparticles are found in expected size. (Figure 8)

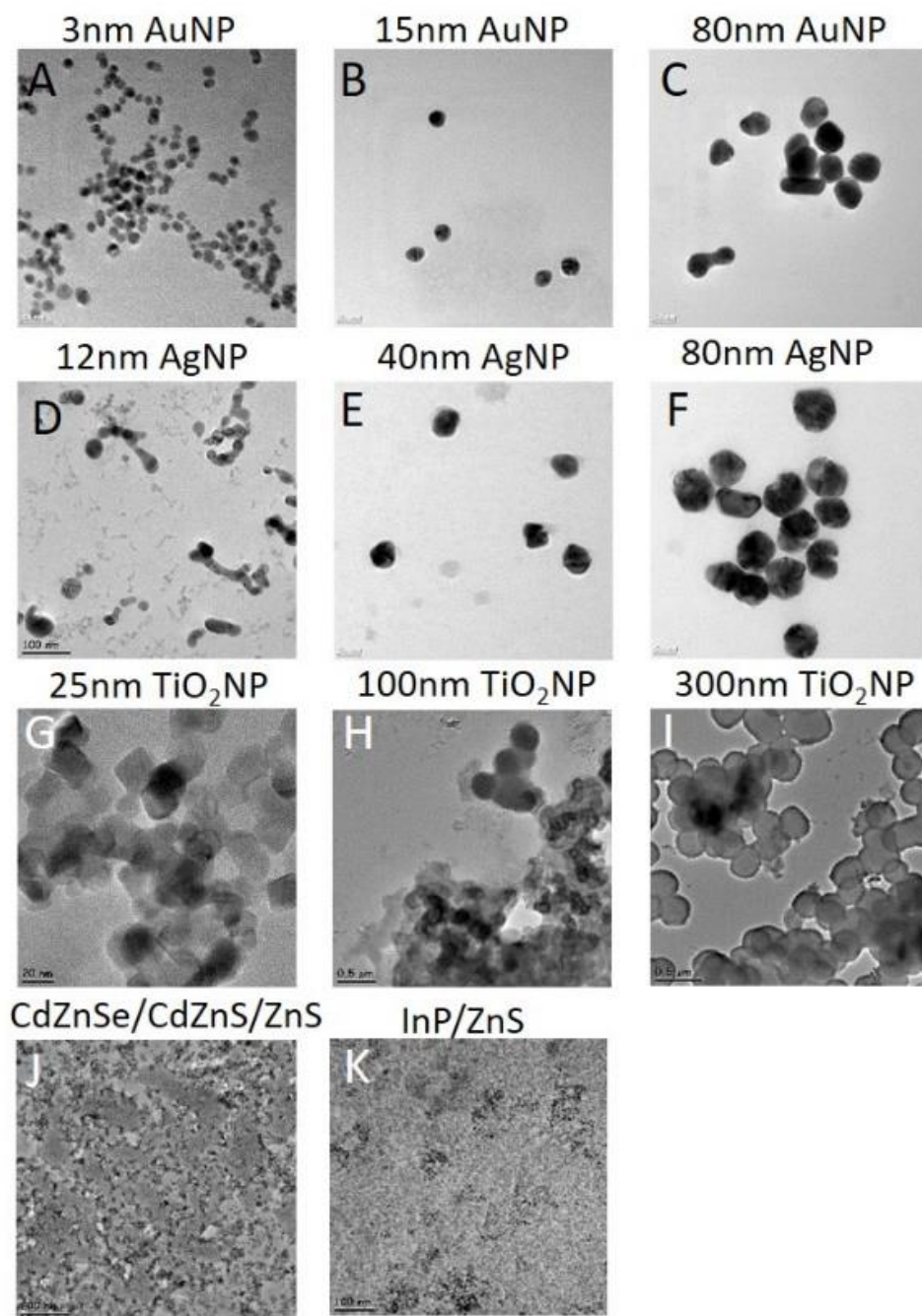


Figure 8: Characterization of nanoparticles with TEM. The TEM images show the distribution and sizes of A) 3nmAuNP B) 15nmAuNP C) 80nmAuNP D) 12nmAgNP E) 40nmAgNP F) 80nmAgNP G) 25nmTiO₂NP H) 100nmTiO₂NP I) 300nmTiO₂NP J) CdZnSe/CdZnS/ZnS and K) InP/ZnS

2.4.3. MTT Assay Results

For toxicity assessment experiments, we must first decide on the toxic concentration of the NPs on HepaRG cells. For this purpose, we conducted MTT assays for the evaluation of the cell viability at different concentrations of NPs. (Figure 9) Varying concentrations of nanoparticles for gold, silver, titanium and quantum dots were added on the cells and they were incubated for 24hr and then the MTT crystals were formed by alive cells. The viability of the cells decreased noticeably as the concentration of the NPs increased.

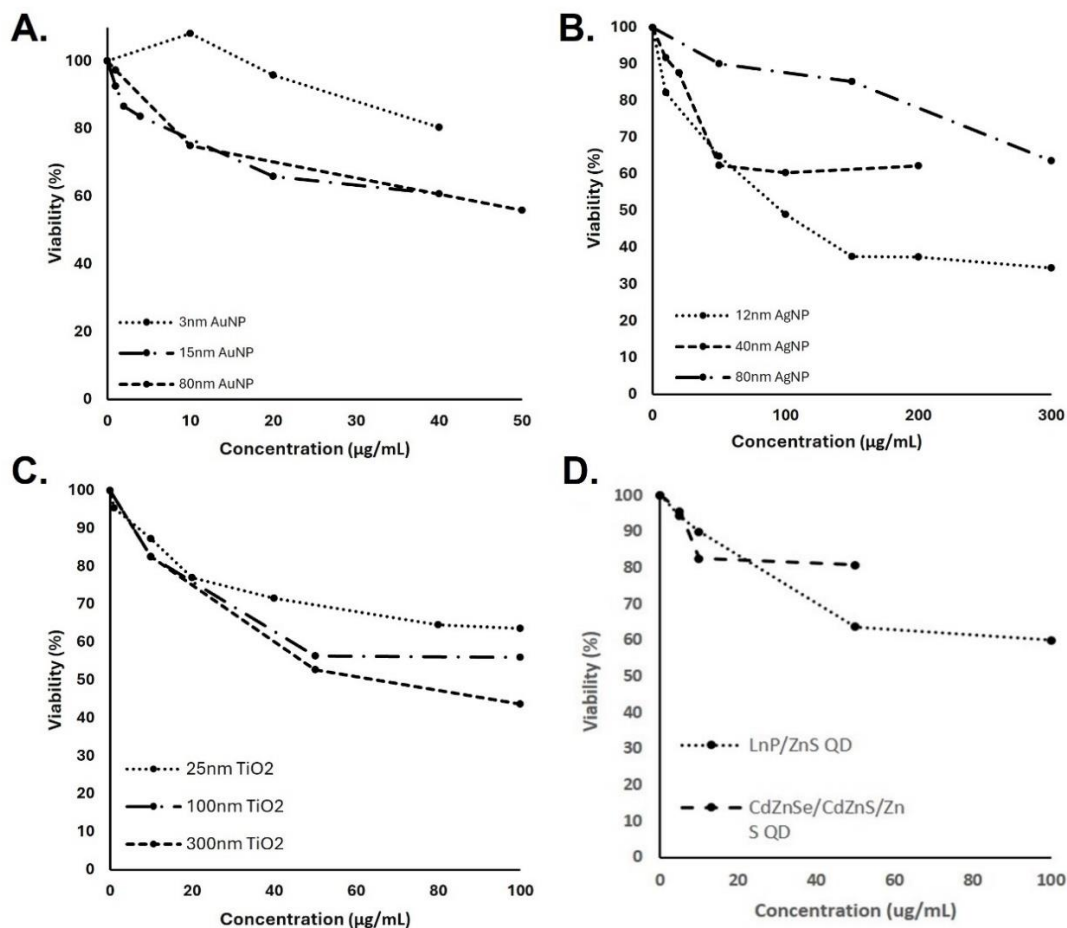


Figure 9: Assessing the viability of the NPs at different concentrations on HepaRG cells.

A) Viability of HepaRG cells that are treated with A) gold nanoparticles B) silver nanoparticles C) titanium nanoparticles D) quantum dots.

For the mRNA isolation for RNA-seq experiments, we decided to use the concentration at which the cell viability is 80%. We chose this concentration because we didn't want the cell viability to be too low since we are going to collect mRNA from them, also we didn't want it to be too high because we want the cells to have a certain amount of cell death which is the implication of the nanoparticle toxicity. For that reason, we used 20µg/mL,

10µg/mL, 10µg/mL for 3nm, 15nm and 80nm AuNP respectively. We used 20µg/mL, 50µg/mL, and 200µg/mL nanoparticles for 12nm, 40nm and 80nm AgNP, respectively. We used 20µg/mL, 10µg/mL, 10µg/mL for TiO₂NPs. And we used 20µg/mL and 60µg/mL for the QDs. Then we continued collecting mRNA from HepaRG cells with these concentrations of nanoparticles.

2.5. Discussion

Nanoparticles, although used in diverse products, knowledge about their toxicity needs further analysis. The mechanism of toxicity is very important for the usage of these particles in medicine and drug delivery applications. Every size, shape and material have their own mechanism to enter the cell, to interact with the cell and to cause toxicity on the cell. That is the reason, every type and size of the nanoparticle should be evaluated in terms of its mechanism of the toxicity on the cells, so that according to their levels of toxicity, we can decide on their size, shape, functionalization and composition for every specific function that will be used. The detailed toxicity mechanism of nanoparticles is a missing piece in the puzzle that restricts the use of these materials effectively. Nowadays, some articles published about the toxicity of the NPs however, they all lack the diverse size and material of the NPs as well as a 3D spheroid system to test on. Spheroids mimic real organs. That is the reason that they give most similar results that are comparable to the real organ, because they provide cellular microenvironment to the cells.

RNA-seq analysis is a high-throughput sequencing technique that gives detailed information about the transcript in the cells. By doing RNA-seq analysis we aim to get more information about the cellular response of the HepaRG cells to the nanoparticle

toxicity. When these cells are incubated with a certain concentration of NPs, cells change their expression profiles, and they respond to toxicity.

2.6. Conclusion

In conclusion, we found that nanoparticles caused a decrease in the cell viability in all materials and sizes. This means they exert a certain level of cellular stress on HepaRG cells. As we see that the size, material and concentration of the nanoparticles changes the viability percentage of the cells, this means every size and every material has its own mechanism of entrance, interaction and toxicity. Generalizing the properties a few nanoparticles to all would be a big mistake. Every material and size should be investigated individually, since their interaction with the cells changes with the physicochemical properties.

CHAPTER 3

Transcriptomic Analysis of Nanoparticle Toxicity

The work described in this section is partially published in the following study [70]:

Merve Erden Tüçer ^a, Nazlıcan Tunç ^a, Suat Tüçer ^a, Rana Acar ^b, Duygu Deniz Usta ^c,
Kouroush Salimi ^d, Özlen Konu ^{a,b}, Urartu Özgür Şafak Şeker ^{a*}

a: UNAM–Institute of Materials Science and Nanotechnology, National
Nanotechnology Research Center, Bilkent University, 06800, Ankara, Turkey

b: Bilkent University, Department of Molecular Biology and Genetics, 06800, Ankara,
Türkiye

c: Gazi University, Faculty of Medicine, Department of Medical Biology and Genetics,
06500, Ankara, Türkiye

d: Ankara Yıldırım Beyazıt University, Faculty of Engineering and Natural Sciences,
Department of Chemical Engineering, 06010, Ankara, Türkiye

*Corresponding Author: Urartu Özgür Şafak Şeker, email: urartu@bilkent.edu.tr

Reprinted with permission of authors from following publications;

<https://doi.org/10.1016/j.cbi.2024.111303>

3.1. Objective of The Study

Transcriptomic analysis is important for investigating the expression profiles of the cells under certain conditions. Nanoparticle toxicity needs to be analyzed at a transcriptomic level because it causes gene expression changes in the cells. Mechanism of the NP toxicity needs further analysis because increasing human exposure to NPs grows the concern about the biosafety of these materials. Nanotoxicology is the area that investigates the toxicity of the nanoparticles. However, until now, many studies about the toxicity of the NPs using the conventional techniques such as MTT assay have some limitations which could be the interference of the nanoparticles with the assay. In this study, to investigate the mechanism of the NP toxicity, we are doing transcriptomic analysis by using high-throughput sequencing technique RNA-seq. By this analysis, we aim to find the genes that respond to NP toxicity. Finding these genes will help us understand the mechanism of the NP toxicity, and bioanalytically assess the toxicity levels of NPs on HepaRG spheroids.

3.2. Introduction

Nanoparticles are foreign particles for the human body and cells. As the human body is exposed to NPs, they circulate in the body and then they enter the cells and accumulate. Increasing exposure with NPs, the human body might surpass the safe doses and cause health problems. [79] Preventing of these detrimental effects depends on the understanding of the mechanism of the toxicity of these NPs. Gene expression profiles of the cells that

are exposed to NPs changes. These changes may represent the mechanism of toxicity and the cellular events that take place upon NP toxicity might be used for managing the negative impacts of NPs.

Next generation sequencing is a high-throughput technique that is used to analyze global expression of the transcriptome. [78] This technique offers better coverage and higher resolution on the transcriptome. Apart from expression analysis, it is used for the discovery of novel transcripts and detection of alternative splicing. The workflow of the RNA-seq sample preparation and data analysis is shown in Figure 10.

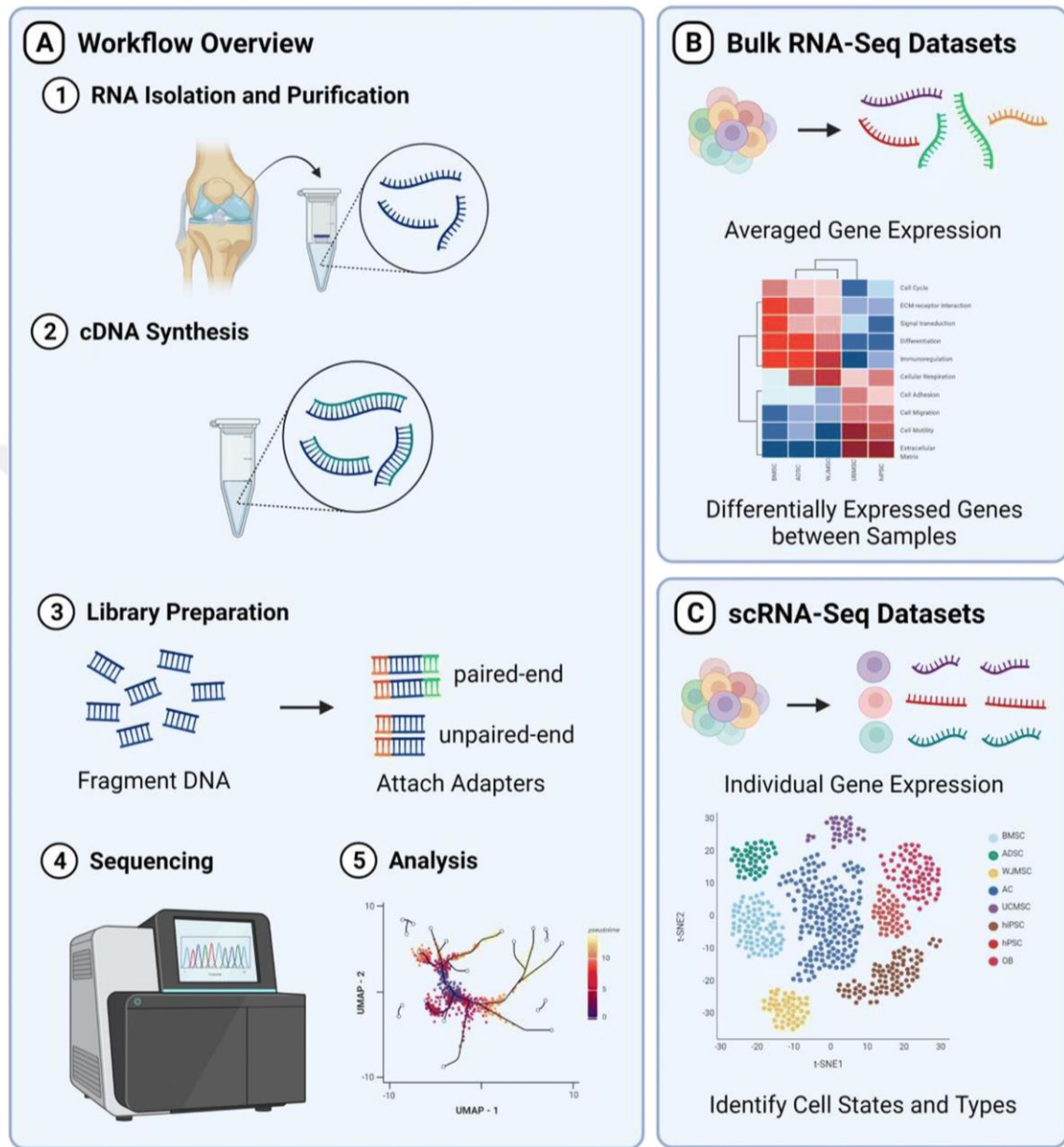


Figure 10: Schematic representation of high-throughput RNA sequencing workflow.

Reprinted with permission from (Thomas et al., 2022) [80]

After RNA-sequencing, raw data analysis is done with special software that is developed for the assembly and quantification of the transcripts. The methodology of the RNA-seq data analysis involves mapping to the reference genome, assembly of the reads to full length genes, quantification of the expression data, and differential expression analysis. (Figure 11) First, quality check of the raw data is done with FAST-QC. [81] Transcriptomic analysis involves the usage of human genome which is sequenced and can be found in databases. Genome alignment of the reads to the human genome is done with hierarchical indexing for spliced alignment of transcripts (HISAT). HISAT is a very efficient aligner as well as, it has a reduced memory requirement. [82] After the alignment of the reads to the human genome, we have to assemble and quantify the reads to find the transcript level expressions. In this step we use StringTie, which is a computational method, to assemble the transcripts and quantify them. [83] StringTie can also do de novo genome assembly as well as mapping reads to the reference genome. After assembling and quantifying the transcripts, Ballgown is used for the differential expression analysis. [84] This package of R, enables flexible analysis of the expression data.

After the differential expression analysis, the genes that changed their expression significantly, will be the ones which responded to NPs toxicity. Pathway enrichment analysis of the differentially expressed genes (DEGs) is needed to find the biological relevance of these changes. Gene ontology (GO) analysis aims to link the DEGs to the biological processes, cellular compartments and molecular functions. Basically, we can understand the function of a DEG with the help of GO analysis. This helps us to understand the toxicity mechanism of the nanoparticles on the cells. Even we can have an idea about the entrance route of the NPs into the cells, because they will enrich the genes that are related to these pathways. [86]

Transcriptomic analysis is an efficient way of investigating the nanotoxicology, because it offers very detailed information about the gene expressions in the cells. By this way, we can predict the changes in the cellular processes upon NP treatment.

3.3. Materials and Methods

3.3.1. Software and Packages

Raw read quality check is done with FAST-QC software. Building genome indexes and raw read genome alignment is done with Hisat2. Conversion of SAM files to BAM file was done with samtools. Genome assembly and estimation of the transcript abundances were done with StringTie. Differential gene expression analysis was done with Ballgown, an R package. Gene ontology (GO) and Hallmark gene set enrichment analysis were done with msigdb package of R. NIH DAVID Bioinformatics tool (<https://david.ncifcrf.gov/tools.jsp>) was used for KEGG pathway analysis.

ComplexHeatmap R package was used for constructing with the gene sets driven from GSEA gene lists.

3.3.2. Quality Check of Raw Data with FASTQC

The quality control of RNA-Seq data was conducted using the FastQC with default parameters. The raw data were loaded as a FastQ file to the FastQC software. This program gives quick information about number of total reads, sequence length, per base sequence quality, per tile sequence quality, per sequence quality scores, per base sequence content, per sequence GC content, per base N content, sequence length distribution, sequence duplication levels, overrepresented sequences, and adapter content.

3.3.3. Read alignment with HISAT

The RNA-seq data was analyzed according to the protocol that was proposed by Pertea et al. [85] RNA-seq analysis starts by mapping raw reads to the human genome. Hisat2 is the software that we use for the mapping step. Hisat2 is used in the Bilkent server. Before beginning the analysis, genome indexes should be build. To build the genome indexes, we must download human genome sequence. Genome sequence (GRCh38.p14) is downloaded from GENCODES (https://www.gencodegenes.org/human/release_46.html) Then by using Hisat2, genome indexes are formed with the following code:

```
hisat2-build -p 16 genome.fa genome_indexes
```

Then, raw reads are mapped to the reference genome by using this index file formed. For every sample the code below is run:

```
hisat2 -threads 8 -dta -x indexes/genome -l input1.fq.gz -2 input2.fq.gz -D output.sam
```

Later, by using samtools, SAM files are converted to BAM files:

```
samtools sort -@ 8 -o outputfile.bam inputfile.sam
```

3.3.4. Transcript assembly and quantification with StringTie

StringTie is used to assemble the reads to full genes. For that we need to prepare a folder that contains a gtf file containing gene annotations and this file could be downloaded from RefSeq database. For the input. label part, only the name of the sample should be written without the file type extension. Eg. For the file named ‘sample.gtf’, the input. label will be ‘sample’. The code used in this step is below:

```
stringtie -p 8 -G genes/genes.gtf output.gtf -l input.label input.bam
```

Then, all the transcripts from all samples that will be analyzed should be merged. For this step we should prepare a txt file that contains the names and extensions of all the files.

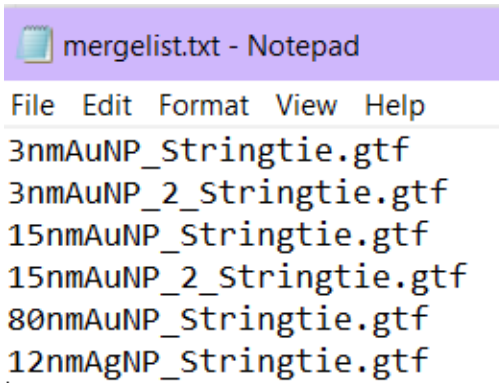


Figure 12: A sample txt file that will be used for the merge step of the StringTie.

The code that will be used to merge all the samples is below:

```
stringtie --merge -p 8 -G genes/genes.gtf -o merged.gtf mergelist.txt
```

For quantifying and estimating the transcript abundances, the code below is used:

```
stringtie -e -B -p 8 -G merged.gtf -o ballgown/output/output.gtf input.bam
```

3.3.5. Differential expression analysis with Ballgown

GTF files that are formed with StringTie will be used in R with the Ballgown package for the differential expression analysis. First, we need to prepare a phenotype data file. Phenotype_data is a csv file that contains sample names and sample conditions. It can be prepared in txt and saved in the csv format. List should be in the same order as in the list.files ("ballgown") comment list.

1	id, type, state			
2	NP12nmAgNP1, Silver, Nanoparticle			
3	NP12nmAgNP2, Silver, Nanoparticle			
4	NP15nmAuNP1, Gold, Nanoparticle			
5	NP15nmAuNP2, Gold, Nanoparticle			
6	NP25nmTiO21, Titanium, Nanoparticle			
7	NP25nmTiO22, Titanium, Nanoparticle			
8	NP3nmAuNP1, Gold, Nanoparticle			
9	NP3nmAuNP2, Gold, Nanoparticle			
10	NP40nmAgNP1, Silver, Nanoparticle			
11	NP40nmAgNP2, Silver, Nanoparticle			
12	NP80nmAgNP1, Silver, Nanoparticle			
13	NP80nmAgNP2, Silver, Nanoparticle			
14	NP80nmAuNP1, Gold, Nanoparticle			
15	NP80nmAuNP2, Gold, Nanoparticle			
16	NPHepaRG1, NT, Control			
17	NPHepaRG2, NT, Control			
18				

Figure 13: A sample phenotype_data file. Columns should be separated with a comma.

Phenotype_data is read by R with the code below:

```
Pheno_data = read.csv ("phenotype_data.csv")
```

Then, expression data is read with the command below:

```
bg = ballgown(dataDir = "ballgown", samplePattern = "NP", pData=pheno_data)
```

If the code above gives an error, make sure that the order of the files in the ballgown folder is in the same order with the names of the samples in the phenotype data file.

```
All (pheno_data$sids == list.files ("ballgown"))
```

During RNA-seq data analysis, we have to get rid of the low-abundance genes. To do that we will positively filter the genes that have an expression value bigger than 1 with the code below:

```
Bg_filt = subset (bg, "rowVars(expr (bg)) > 1",  
                  genomesubset=TRUE
```

Calculation of the p-values and finding the genes that are differentially expressed we use the code below:

```
Results_transcripts= stattest (bg_filt, feature="transcript",  
                               covariate="state", getFC=TRUE, meas="FPKM")
```

Finding the differentially expressed genes:

```
Results_genes= stattest (bg_filt, feature="gene", covariate="state",  
                          getFC=TRUE, meas="FPKM")
```

Adding gene names and gene IDs to the transcript data frame:

```
results_transcripts=  
data.frame(geneNames=ballgown::geneNames(bg_chrX_filt),  
           geneIDs=ballgown::geneIDs(bg_chrX_filt), results_transcripts)
```

Sort the results from the smallest P value to the largest

```
results_transcripts = arrange(results_transcripts,pval)  
results_genes = arrange(results_genes,pval)
```

Write the results as a csv file

```
write.csv(results_transcripts, "transcript_results.csv", sep=";", "dec="",  
row.names=FALSE)  
  
write.csv(results_genes, "chrX_gene_results.csv", sep=";", "dec="",  
row.names=FALSE)
```

Identify transcripts and genes with a q value <0.05

```
subset(results_transcripts, results_transcripts$qval<0.05)  
  
subset(results_genes, results_genes$qval<0.05)
```

3.3.6. Statistical Analysis

For the determination of the DEGs for each NP, the p-value that is calculated by Ballgown by comparing the control (NP not added) and the test group (NP added) is used. DEGs that have the $pval < 0.05$ and the $|\log(\text{fold change})| > 0.5$ were determined to be statistically significant. Gene ontology analysis and KEGG pathway analysis are done and the significant terms are determined according to the P-value (< 0.05).

3.3.7. Gene Ontology/Hallmark and KEGG Pathway Analysis

Gene ontology and KEGG pathway analysis are done in National Institutes of Health DAVID Bioinformatics (<https://david.ncifcrf.gov/tools.jsp>) website. Hallmark gene expression analysis was conducted with msigdb package of R.

3.3.8. Plots

Volcano plots were plotted with ggplot, venn diagrams were plotted VennDiagram and RColorBrewer. Heatmaps were plotted with ComplexHeatmap in R.

3.4. Results

3.4.1. Quality Check and Alignment Scores

Quality check of the raw RNA-seq data is done by Fast-qc software. This software gives the number clean reads, lengths of the reads, GC percentage of the reads. These numbers are important for the judgement of quality of the raw data. When raw data is aligned to the human genome by Hisat2, at the end of the run, Hisat2 gives the percentage of the alignment, which also shows the percentage of the reads that is aligned to the genome, so quality of the sequence data. The quality scores are given in a table. (Table 1)

Quality of the RNA-seq data seems to be very good, because number of the clean reads is higher than 20 M and their GC percentage is near to 50%. Sequence length is 150bp for every sample which is enough for a qualified transcriptomic analysis. Also, alignment percentages of the raw data to the human genome are greater than 90% for every sample, which is a great score.

Table 1: Summary of the quality check of the RNA-seq raw data.

Samples	Clean Reads	GC%	Sequence Length (bp)	Alignment Rate (%)
HepaRG-1	22,648,417	49	150	94.88
HepaRG-2	20,619,624	51	150	95.42
3nmAuNP-1	23,423,763	51	150	94.51
3nmAuNP-2	22,334,044	51	150	95.13
15nmAuNP-1	20,639,099	49	150	95.81
15nmAuNP-2	20,657,077	51	150	95.35
80nmAuNP-1	20,097,428	52	150	95.60
80nmAuNP-2	22,358,613	52	150	92.90
12nmAgNP-1	40,387,339	50	150	93.90
12nmAgNP-2	20,823,378	50	150	95.05
40nmAgNP-1	21,389,604	51	150	95.12
40nmAgNP-2	22,612,895	51	150	94.93
80nmAgNP-1	22,650,606	50	150	95.04
80nmAgNP-2	21,203,308	50	150	94.66
25nmTiO ₂ NP-1	22,651,663	50	150	94.20
25nmTiO ₂ NP-2	21,274,761	50	150	93.46
100nmTiO ₂ NP-1	22,126,102	49	150	92.89
100nmTiO ₂ NP-2	20,076,735	50	150	92.91
300nmTiO ₂ NP-1	22,645,545	50	150	92.07
300nmTiO ₂ NP-2	22,407,754	47	150	93.11
QD1-1	26,753,619	51	150	92.47
QD1-2	26,120,473	50	150	93.80
QD2-1	20,746,831	49	150	94.12
QD2-2	21,908,236	49	150	93.90

3.4.2. Gene Expression Profiles of Nanoparticle Treated Spheroids

After differential gene expression analysis, the genes that have a p-value smaller than 0.5 and also have a $|\log(\text{FC})|$ smaller than 0.5 are chosen to be significant. According to the results, the DEGs were plotted with a volcano plot with the values $-\log(\text{p-val})$ on the y-axis and $\log(\text{FC})$ on the x-axis. (Figure 14) Significantly upregulated genes are shown with red dots, significantly downregulated genes are shown with blue dots, and nonsignificant genes are shown with grey dots. According to the results, 3nm AuNP has 370 DEGs, 213 upregulated and 157 genes downregulated. 15nm AuNP has 116 DEGs, 68 upregulated and 48 downregulated. 80nm AuNP has 205 DEGs, 112 upregulated and 83 downregulated. 12nm AgNP has 178 DEGs, 118 upregulated and 60 downregulated. 40nm AgNP has 977 DEGs, 460 upregulated and 517 downregulated. 80nm AgNP has 711 DEGs, 279 upregulated and 432 downregulated. 25nm TiO₂NP has 538 DEGs, 179 upregulated and 359 downregulated. 100nm TiO₂NP has 366 DEGs, 218 upregulated and 148 downregulated. 300nm TiO₂NP has 660 DEGs, 287 upregulated and 373 downregulated. CdZnSe/CdZnS/ZnS (QD1) has 692 DEGs, 314 upregulated and 378 downregulated. InP/ZnS (QD2) has 1904 DEGs, 811 upregulated and 1093 downregulated.

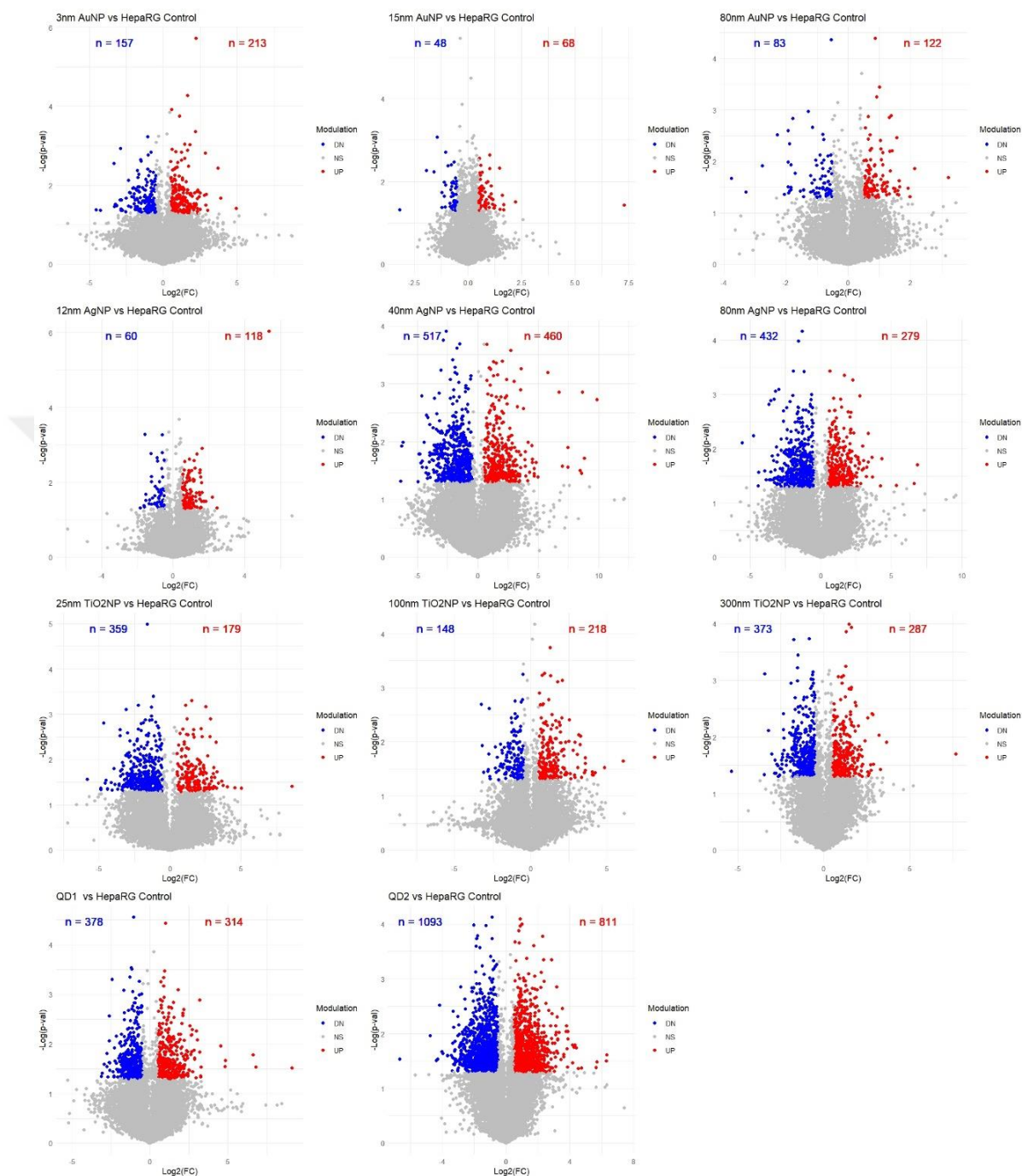


Figure 14: Gene expression profiles of nanoparticle treated HepaRG spheroids. Upregulated DEGs (red), downregulated DEGs (blue) and non-significant genes (grey).

Number of common genes between nanoparticles are shown as Venn diagrams in Figure 15. According to the results we see that there are the greatest number of common genes in quantum dot treated cells.

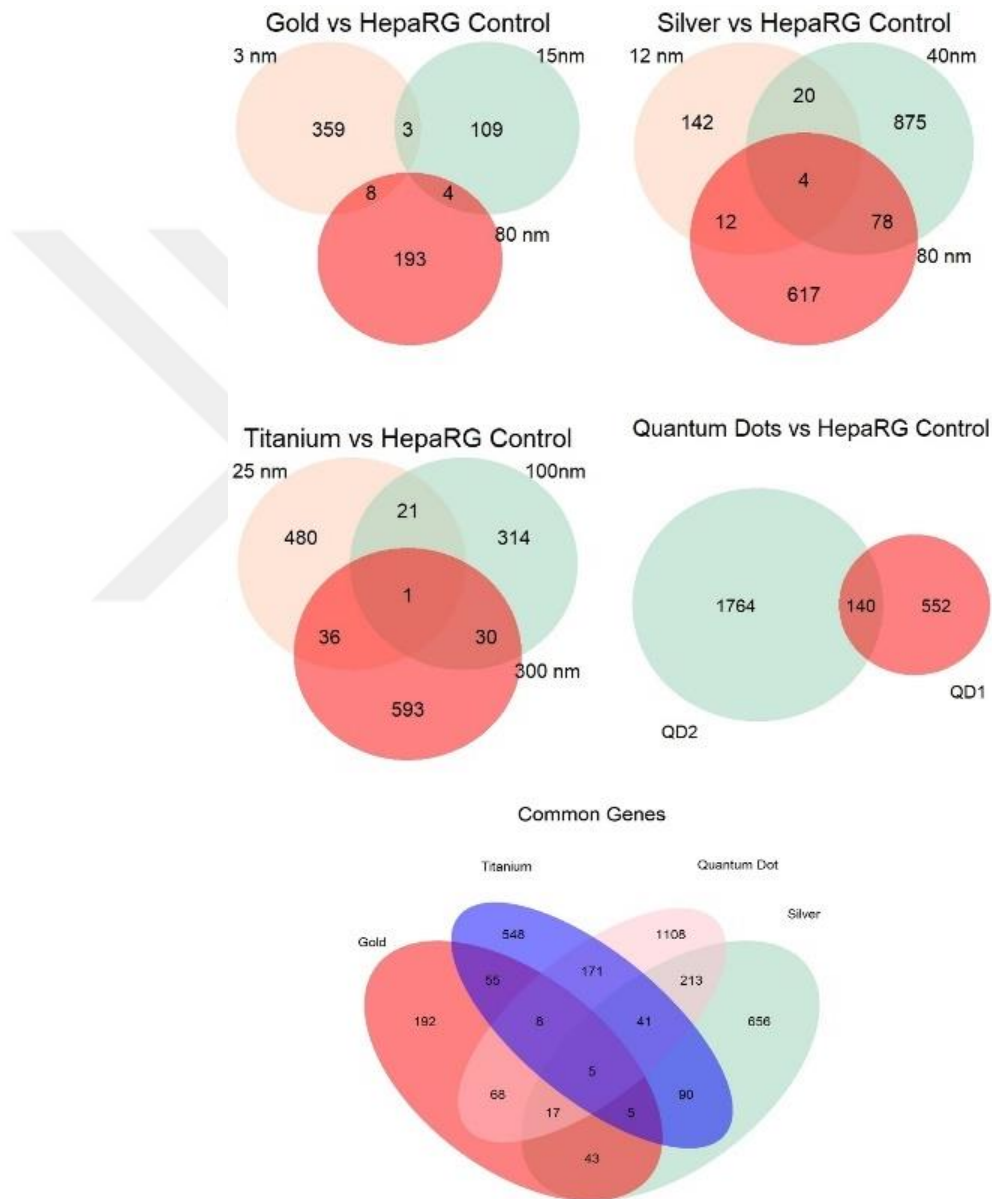


Figure 15: Venn diagrams of the DEGs for each size and material of the NPs. Common genes between different NPs are also shown.

DEGs from each material are combined. 5 common genes were found that were differentially expressed in all four materials of NPs, which are described in **Table 2**. Functions of these genes are related to tumor-promoting, inflammation and EMT.

Table 2: Common genes among all 5 NPs are listed with FC and their cellular function.

Gene Name	Gold FC	Silver FC	Titanium FC	QD FC	Function
IMPA2	2.16	2.37	2.87	2.13	Oncogene[87], [88]
AC078850.1	1.88	3.26	4.61	4.60	Pyroptosis, Inflammation[89]
OPN3	1.42	1.71	2.52	1.63	Epithelial-Mesenchymal Transition and Metastasis[90]
HS3ST3B1	2.82	2.89	4.64	3.89	Epithelial-Mesenchymal Transition[91], [92]
NUP85	1.44	1.75	1.55	3.00	Inflammation, liver disease[93]

3.4.1. Functional Analysis of DEGs

3.4.1.1. Gene Ontology Analysis

After finding the DEGs for each NP, we needed to do gene ontology analysis to relate every gene to a biological process, cellular compartment, or molecular function. Because relating the DEGs to these processes enables us to better understand how the cells respond to NP toxicity and also we will have a better estimate of the mechanism of the toxicity of the NPs on HepaRG spheroids. Gene ontology analysis is done in David.org. (Figure 16-19)

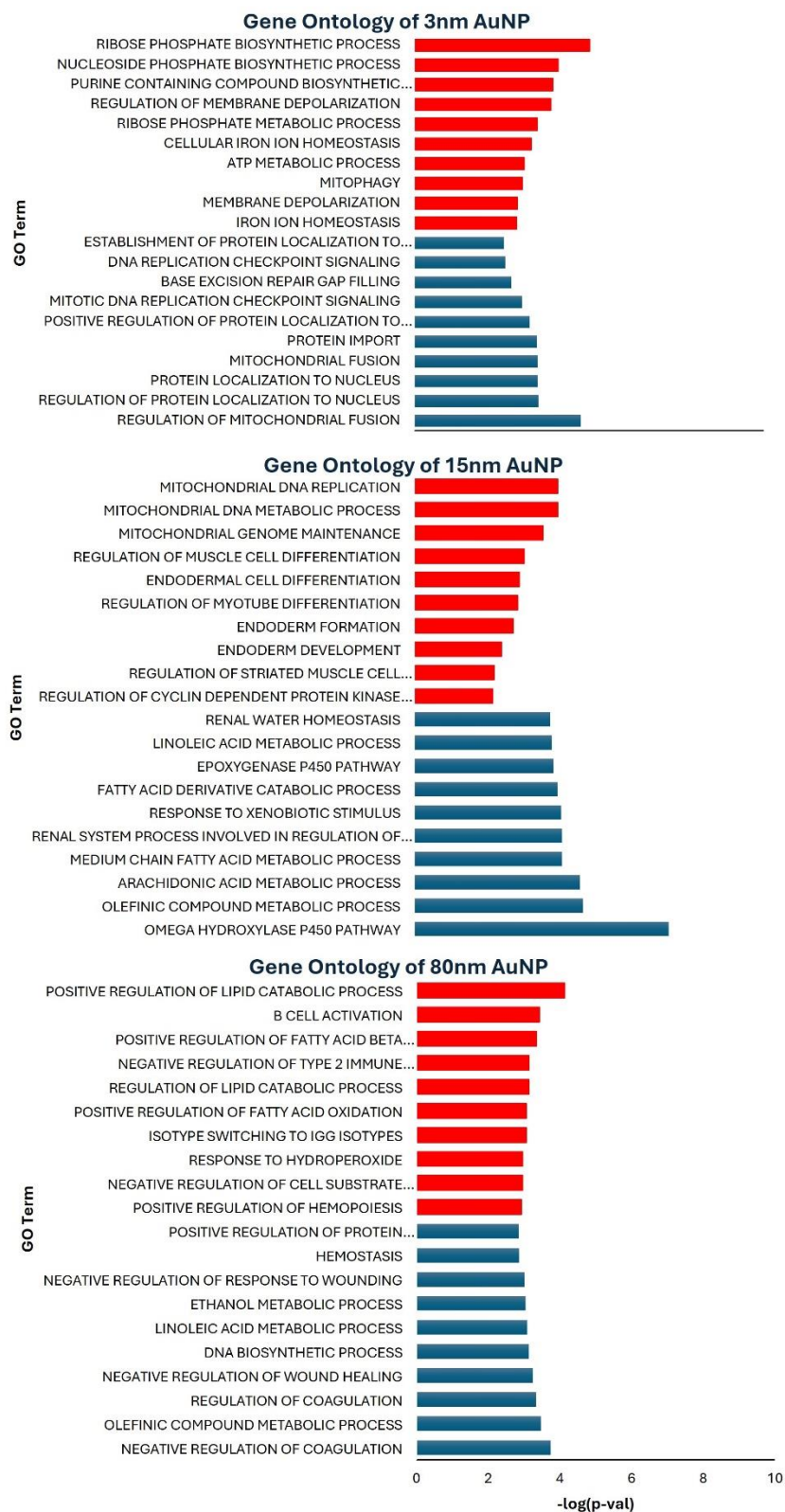


Figure 16: Gene ontology analysis of the DEGs of AuNP treated cells.

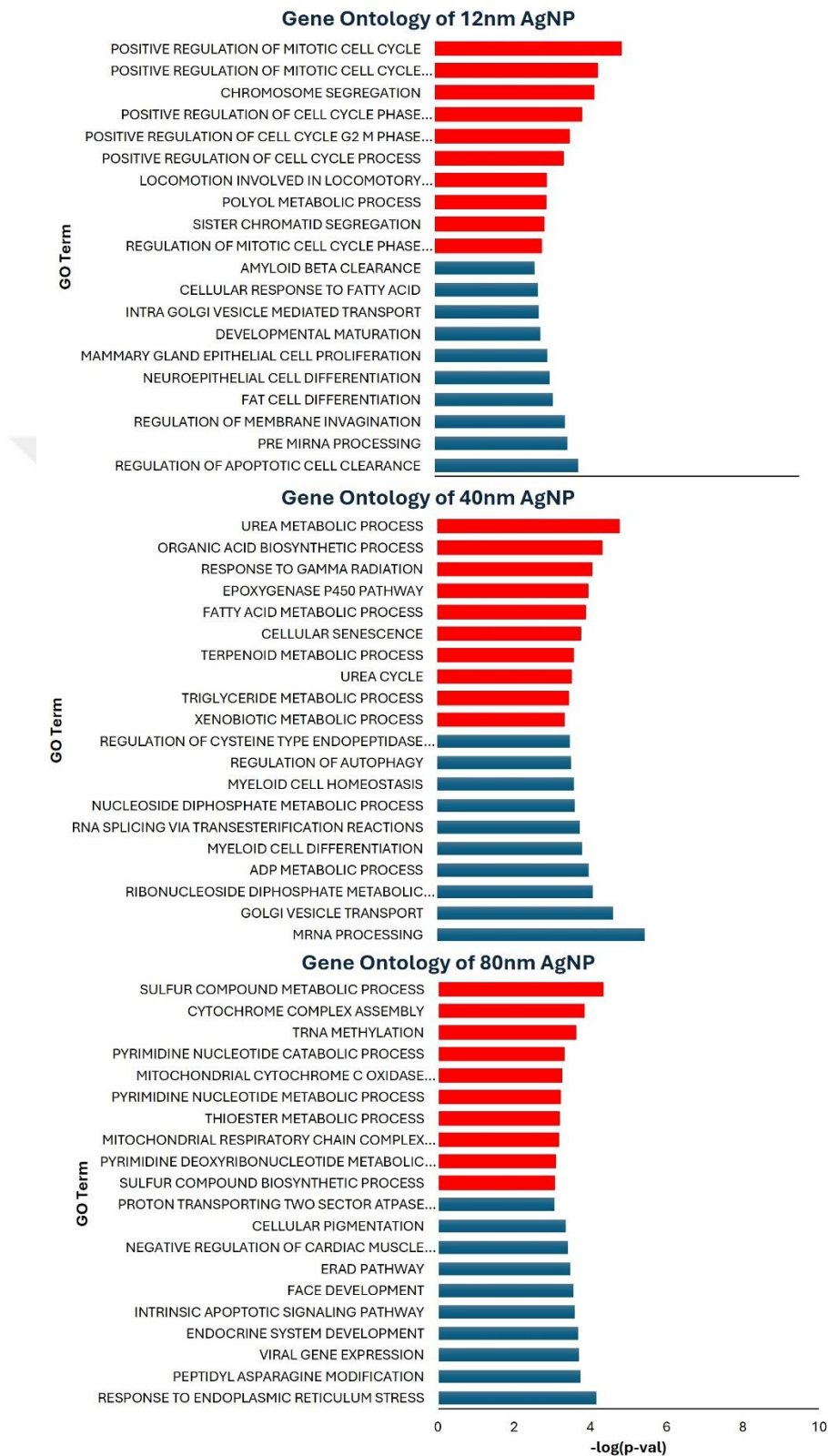


Figure 17: Gene ontology analysis of DEGs of AgNP treated cells.

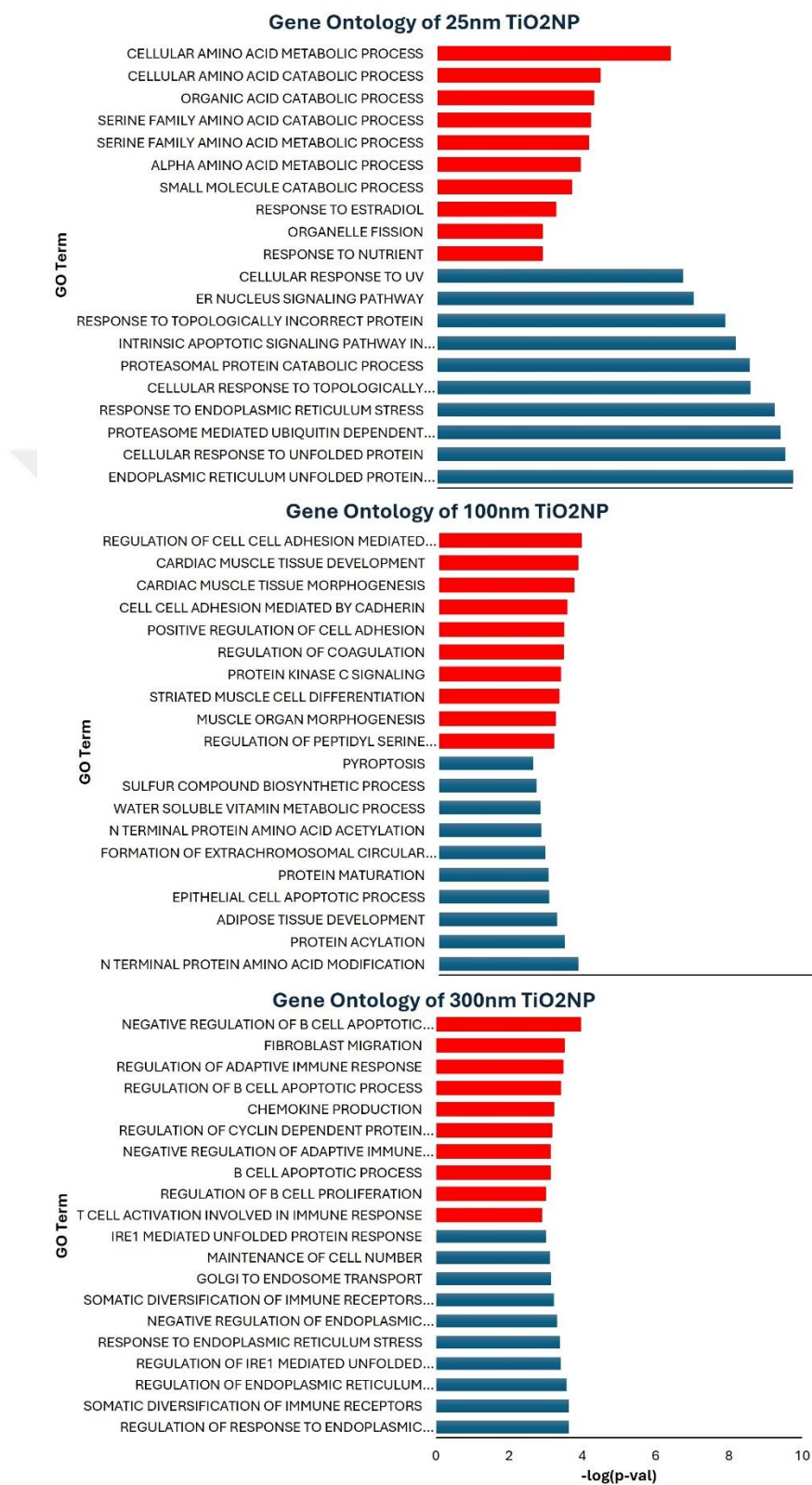


Figure 18: Gene ontology analysis of DEGs of TiO₂NP treated cells.

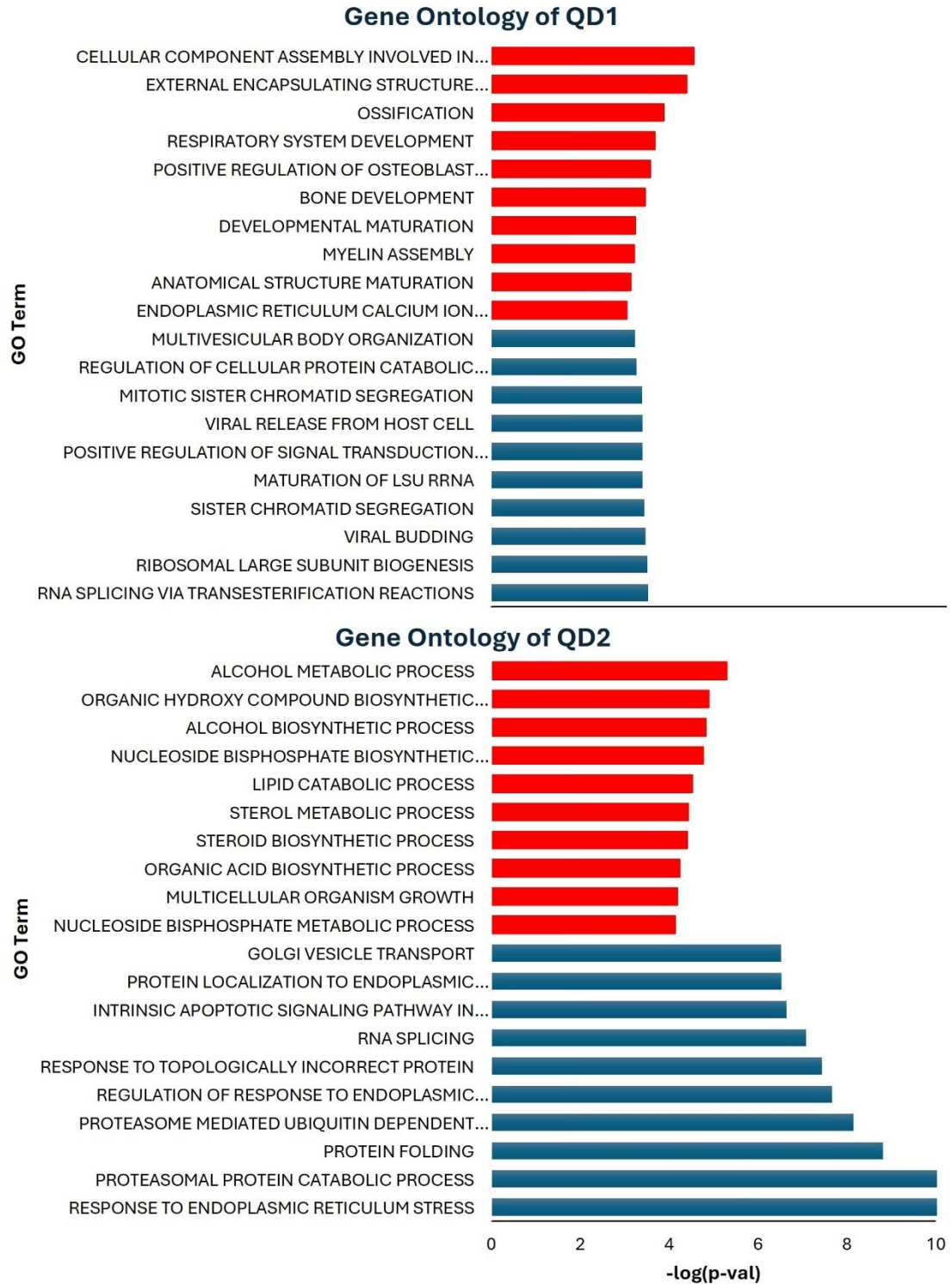


Figure 19: Gene ontology analysis of DEGs of QD treated cells.

3.4.1.2. Hallmark Gene Set Analysis

Hallmark gene set analysis is done with the DEGs for each NP. In transcriptomic analysis many DEGs are found as a result, and these DEGs have to be related to specific pathways or functions in order to make interpretations about the toxicity mechanism of the nanoparticles on HepaRG cells. Therefore, DEGs are subjected to gene set enrichment analysis (GSEA). Most dysregulated gene sets are found and shown in the (Fig. 20). DEGs showed enrichment in the pathways that shows the toxicity mechanism of NPs which are heme metabolism, EMT, stress responses, tumor suppressor pathways, inflammatory and antioxidant pathways, and apoptosis.

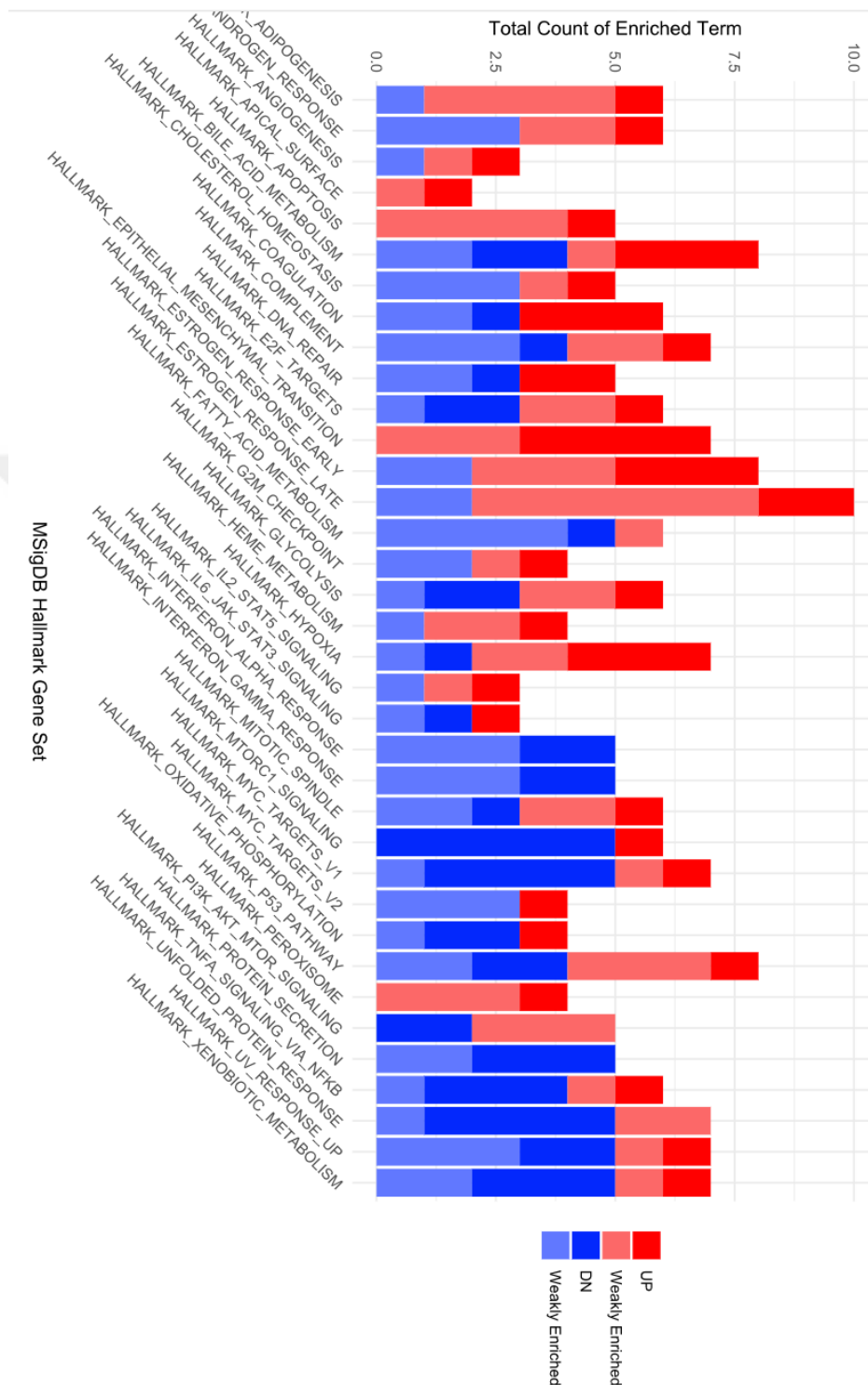


Figure 20: Hallmark gene set analysis of the DEGs.

3.4.1.3. KEGG Pathway Analysis

DEGs for each NP were subjected to Kyoto Encyclopedia of Genes and Genomes (KEGG) pathway analysis to find the functional pathways that they are involved in. (Table 3) DEGs for each NP enriched pathways that are related to various diseases and disorders including hepatitis, rheumatoid arthritis, cancer, Parkinson, Huntington. Also biosynthesis of amino acids was enriched for some of the NPs, which shows the expression of some proteins that functions in the metal homeostasis system.

Table 3: KEGG pathway analysis of the DEGs for each nanoparticle.

NP	TERM	COUNT	PATHWAY ID	P-VALUE
3nm AuNP	Ferroptosis	4	hsa04216	1.109E-02
	Ubiquitin mediated proteolysis	6	hsa04120	2.056E-02
15nm AuNP	Metabolic pathways	12	hsa01100	2.891E-03
	Tyrosine metabolism	3	hsa00350	4.928E-03
80nm AuNP	Complement and coagulation cascades	4	hsa04610	1.334E-02
	Fatty acid degradation	3	hsa00071	2.443E-02
	Type II diabetes mellitus	3	hsa04930	2.883E-02
	Pyruvate metabolism	3	hsa00620	2.883E-02
	Coronavirus disease - COVID-19	5	hsa05171	4.496E-02
12nm AgNP	Fanconi anemia pathway	4	hsa03460	2.863E-03
	Cell cycle	5	hsa04110	8.950E-03
	Human T-cell leukemia virus 1 infection	5	hsa05166	2.815E-02
40nm AgNP	Biosynthesis of amino acids	9	hsa01230	2.099E-03
	Biosynthesis of cofactors	13	hsa01240	2.719E-03
	Aminoacyl-tRNA biosynthesis	8	hsa00970	4.074E-03
	Vibrio cholerae infection	7	hsa05110	4.605E-03

	Epithelial cell signaling in Helicobacter pylori infection	8	hsa05120	6.108E-03
	Metabolic pathways	65	hsa01100	8.331E-03
	Rheumatoid arthritis	9	hsa05323	8.390E-03
	Collecting duct acid secretion	5	hsa04966	1.016E-02
	Glycolysis / Gluconeogenesis	7	hsa00010	1.697E-02
	Amyotrophic lateral sclerosis	20	hsa05014	2.097E-02
80nm AgNP	Protein processing in endoplasmic reticulum	11	hsa04141	1.824E-03
	Nucleocytoplasmic transport	7	hsa03013	1.910E-02
	Human T-cell leukemia virus 1 infection	10	hsa05166	3.094E-02
	Hepatitis C	8	hsa05160	3.648E-02
	Acute myeloid leukemia	5	hsa05221	4.385E-02
	Autophagy - animal	8	hsa04140	4.807E-02
25nm TiO₂NP	Protein processing in endoplasmic reticulum	15	hsa04141	3.207E-06
	Thyroid cancer	5	hsa05216	4.784E-03
	Glioma	6	hsa05214	1.358E-02
	Epstein-Barr virus infection	10	hsa05169	1.369E-02
	Viral carcinogenesis	10	hsa05203	1.452E-02
	Nucleocytoplasmic transport	7	hsa03013	1.547E-02
	Human T-cell leukemia virus 1 infection	10	hsa05166	2.372E-02
	Endometrial cancer	5	hsa05213	2.406E-02
	Longevity regulating pathway	6	hsa04211	2.629E-02
	Hepatitis C	8	hsa05160	2.903E-02
100nm TiO₂NP	EGFR tyrosine kinase inhibitor resistance	4	hsa01521	4.526E-02
	Fluid shear stress and atherosclerosis	5	hsa05418	4.178E-02
	Ubiquitin mediated proteolysis	5	hsa04120	4.289E-02
	MAPK signaling pathway	7	hsa04010	4.807E-02
300nm TiO₂NP	Glycine, serine and threonine metabolism	6	hsa00260	7.203E-04
	Metabolic pathways	44	hsa01100	1.309E-03
	Carbon metabolism	7	hsa01200	1.659E-02
	p53 signaling pathway	5	hsa04115	4.322E-02

	RNA degradation	5	hsa03018	4.874E-02
QD1	Alzheimer disease	22	hsa05010	2.610E-04
	Pathways of neurodegeneration - multiple diseases	24	hsa05022	7.355E-04
	Oxidative phosphorylation	11	hsa00190	1.306E-03
	Parkinson disease	16	hsa05012	1.491E-03
	Prion disease	16	hsa05020	1.917E-03
	Amyotrophic lateral sclerosis	19	hsa05014	2.285E-03
	Butanoate metabolism	5	hsa00650	3.187E-03
	Valine, leucine and isoleucine degradation	6	hsa00280	4.767E-03
	Huntington disease	16	hsa05016	5.519E-03
QD2	mTOR signaling pathway	9	hsa04150	2.978E-02
	Protein processing in endoplasmic reticulum	32	hsa04141	2.022E-07
	Aminoacyl-tRNA biosynthesis	18	hsa00970	8.854E-07
	Selenocompound metabolism	8	hsa00450	5.939E-05
	Protein export	10	hsa03060	1.267E-04
	AMPK signaling pathway	21	hsa04152	1.409E-04
	mTOR signaling pathway	23	hsa04150	7.226E-04
	Human T-cell leukemia virus 1 infection	29	hsa05166	7.994E-04
	Proteasome	10	hsa03050	2.833E-03
	Prion disease	31	hsa05020	5.583E-03
	Longevity regulating pathway	14	hsa04211	6.192E-03

3.4.2. Gene Set Enrichment Analysis

3.4.2.1. Liver Specific Genes

Liver is the main organ in the body for the detoxification of the foreign and harmful substances. It does this by transforming lipophilic substances into more water-soluble

substances, so that they can be effectively discard from the body. [94] Cytochrome p450 system of the enzymes are responsible with the detoxification and metabolism of the xenobiotics and drugs. 57 isoenzymes are found to function in CYP system, but only 6 of them are capable for the detoxification of 90% of the drugs. [95] These enzymes comprise CYP1A2, CYP2C9, CYP2C19, CYP2D6, CYP2E1 and CYP3A4. From the analysis of the DEGs, we see that some of these enzymes are differentially expressed in NP treated cells. (Figure 21)

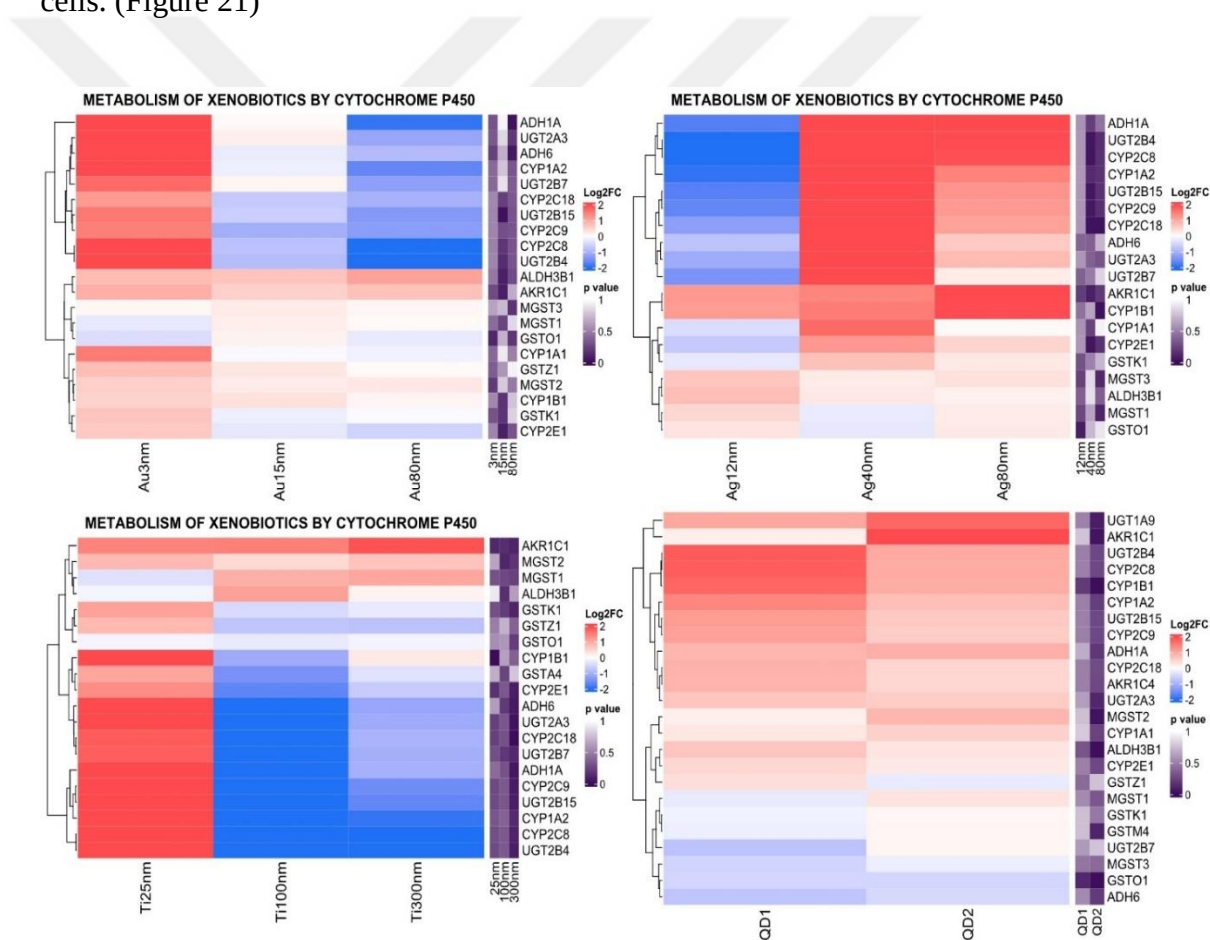


Figure 21: Heatmaps of the CYP enzymes in the NP treated cells.

3.4.2.2. Metal Ion Response Genes

Metal ions cause metal ion response activation in the cells. Because cells want to detoxify these metals since they cause the breakdown of the homeostasis of the cells. Metal ion homeostasis is secured with changes in the gene expression. Metal ion response genes were investigated whether they are differentially expressed. Some SLC transporters, metal ion responsive genes such as metallothionein proteins are differentially expressed in NP treated cells. (Figure 22) MT1A and MT1E are the genes that their expressions increase upon metal ions. [96]

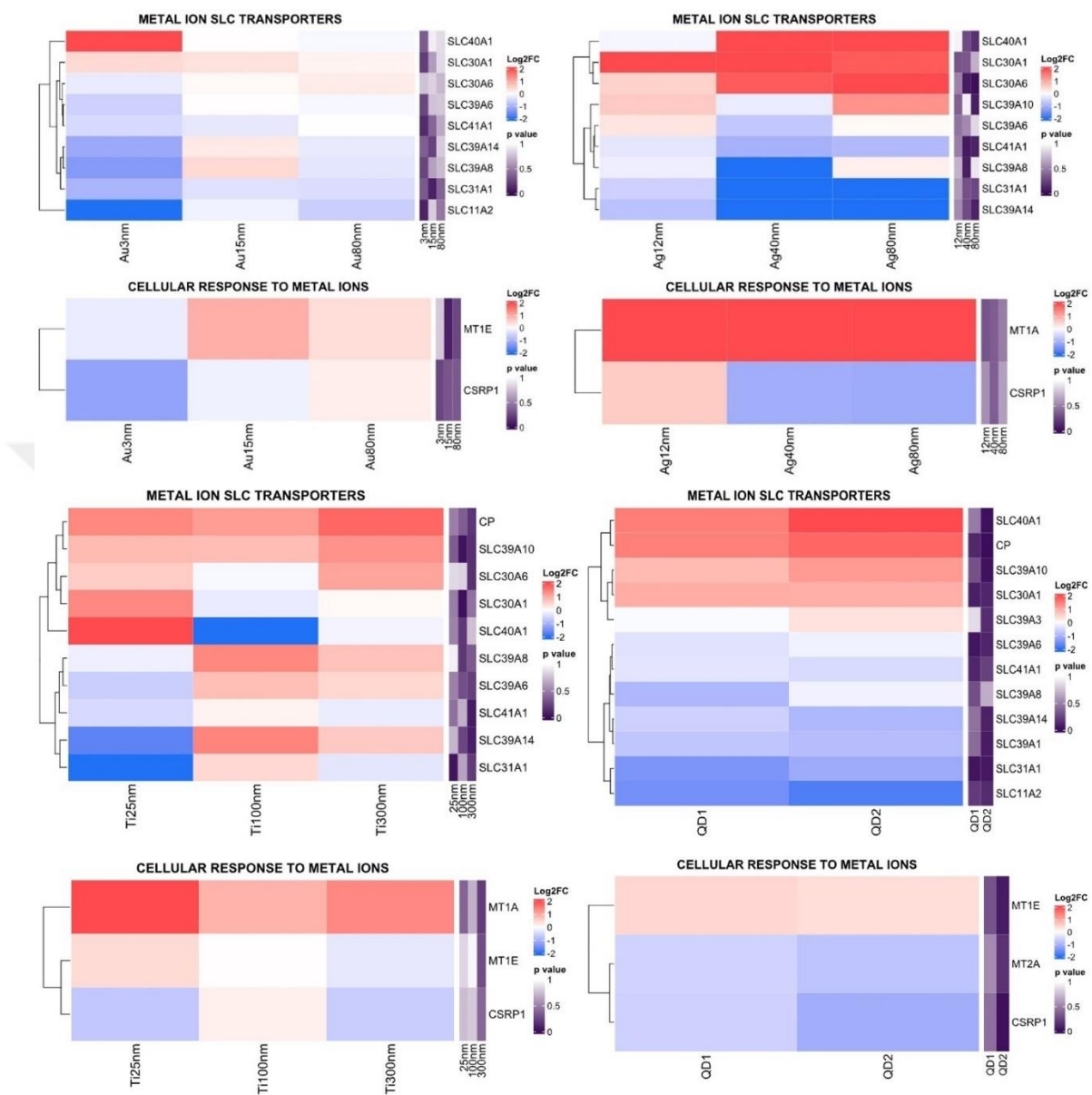


Figure 22: Heatmaps of the metal ion responsive genes.

3.4.2.3. Oxidative Stress Response Genes

Nanoparticle toxicity causes production of reactive oxygen species (ROS) and it causes oxidative stress. Oxidative stress responsive genes are found to be significantly deregulated upon nanoparticle treatment. (Figure 23) The genes that are responsive to the oxidative stress are often antioxidant genes.

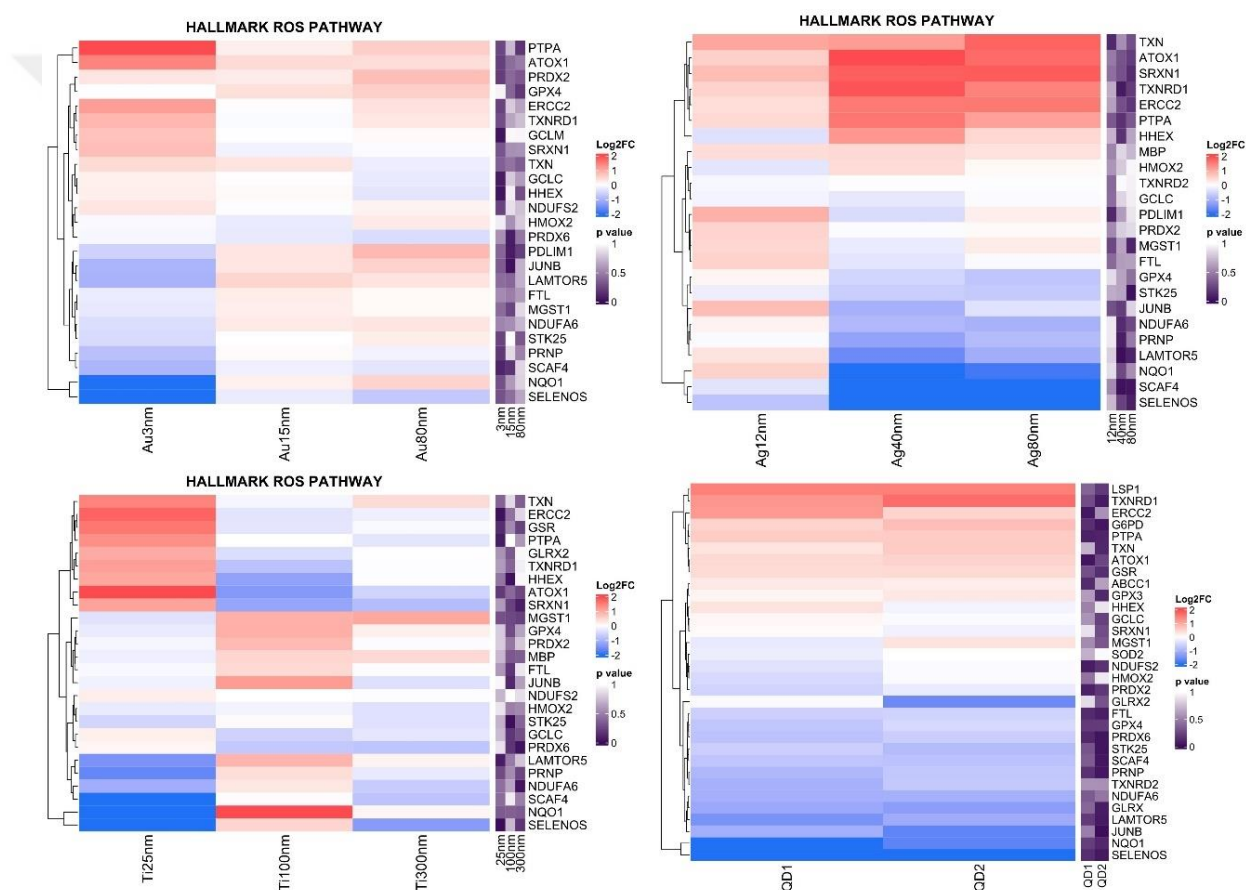


Figure 23: Heatmaps of the oxidative stress responsive genes.

3.4.2.4. Apoptotic Genes

Nanoparticles cause genotoxicity and membrane damage which may cause cell death and apoptosis. [97] [98] For finding the expression profiles of the apoptotic genes, we downloaded apoptotic gene sets from GSEA database, and constructed heatmaps for the apoptotic genes with their p-values. According to our results apoptotic genes were differentially expressed under NP toxicity. (Figure 24-27)

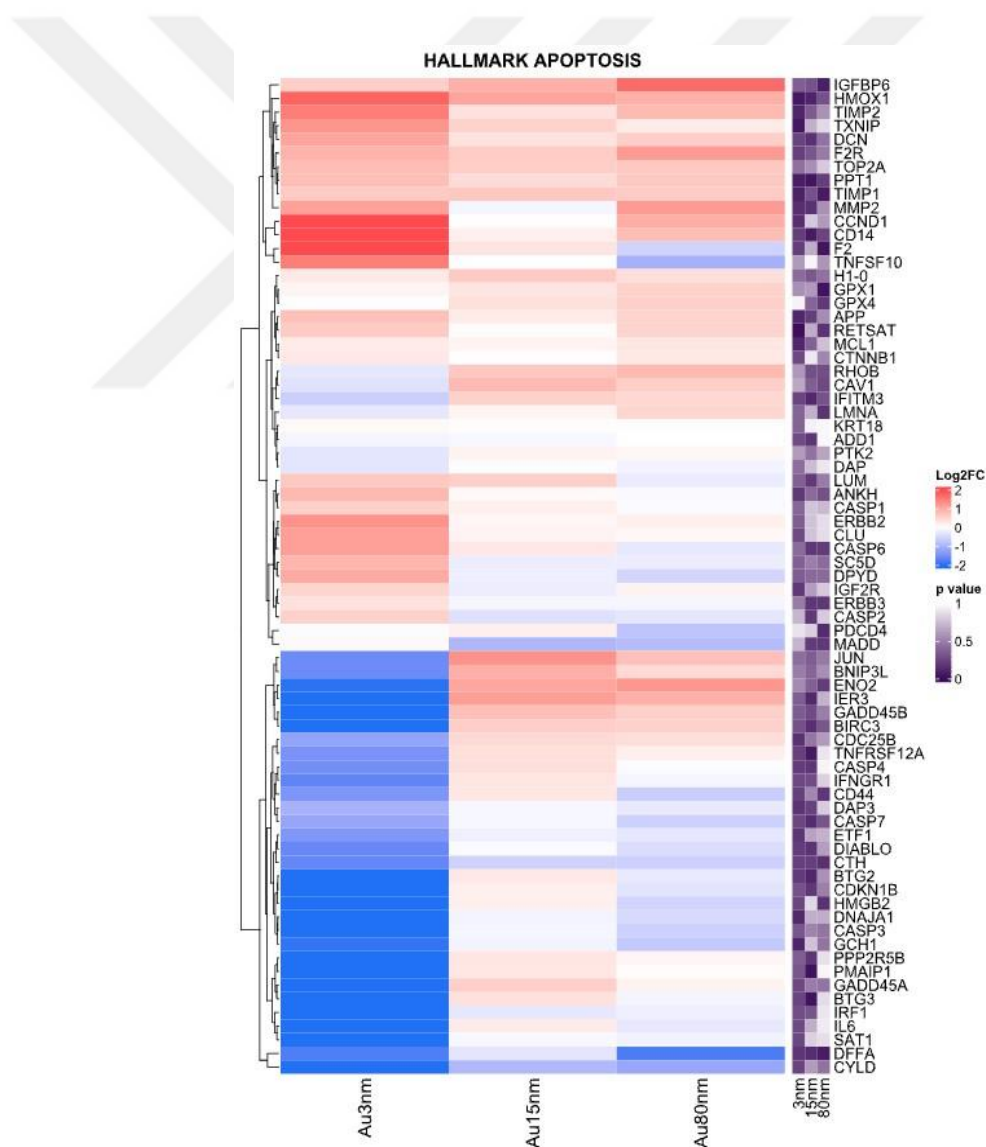


Figure 24: Heatmap of the apoptotic genes for gold nanoparticles.

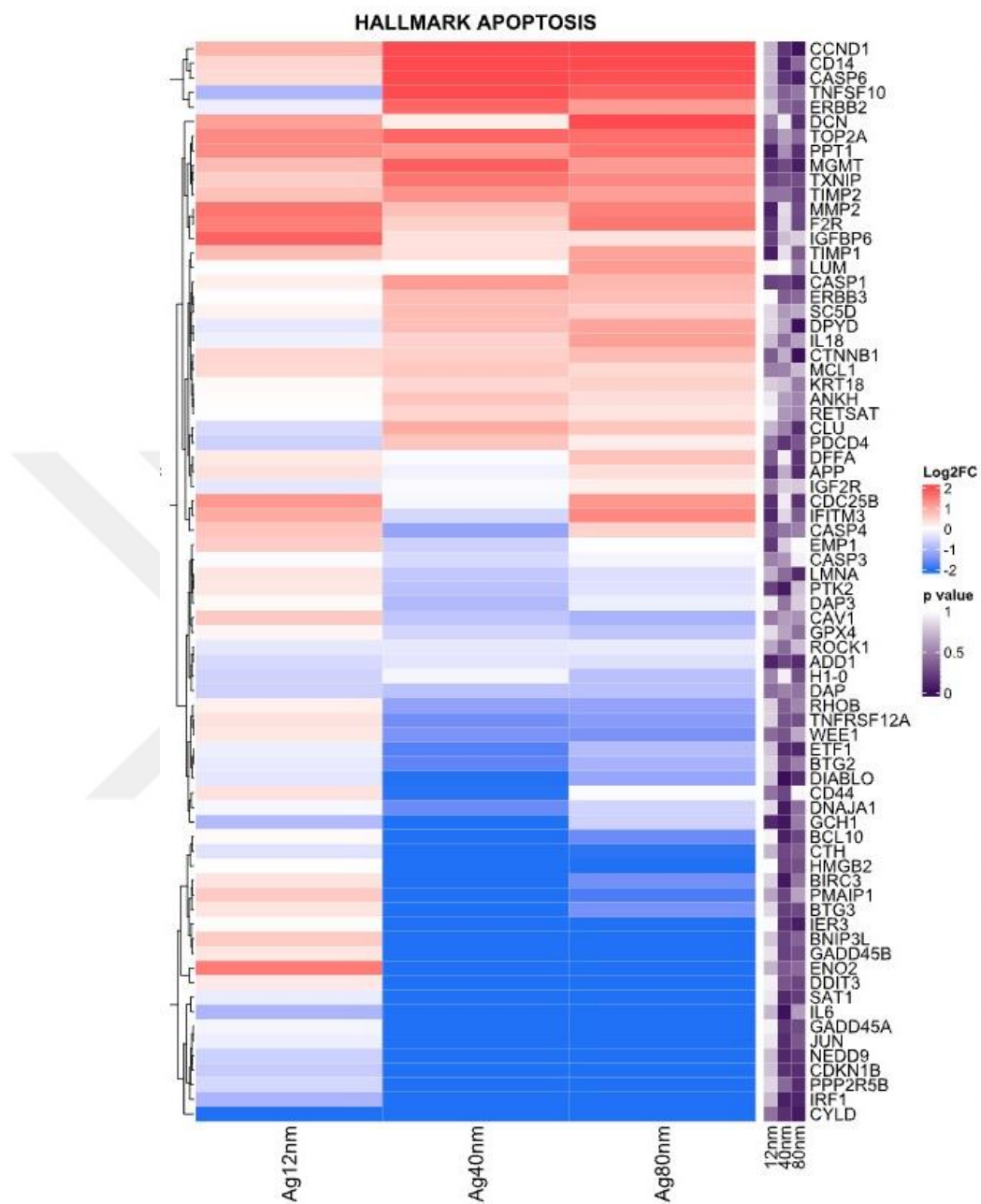


Figure 25: Heatmap of the apoptotic genes for silver nanoparticles.

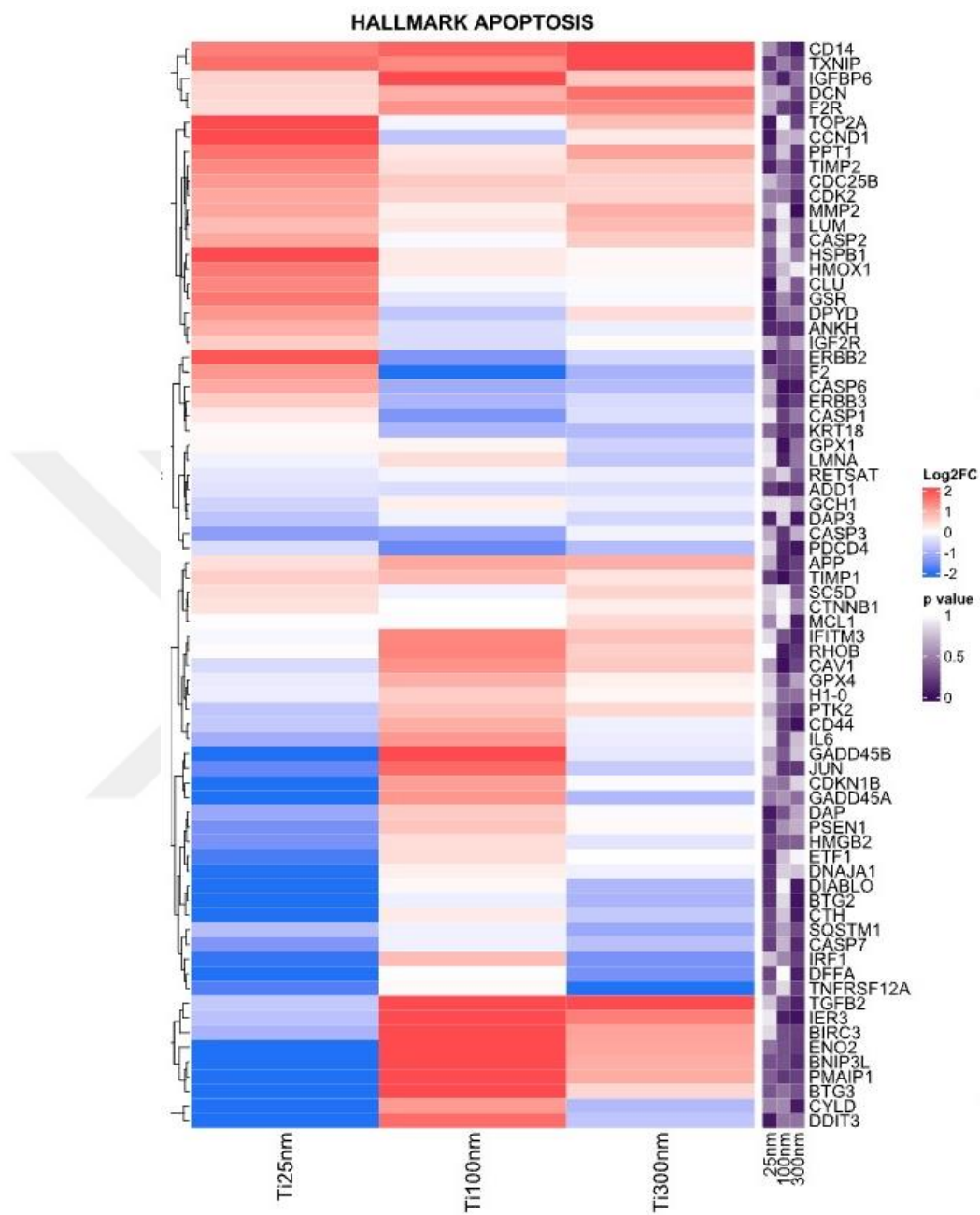


Figure 26: Heatmap of the apoptotic genes for titanium nanoparticles.

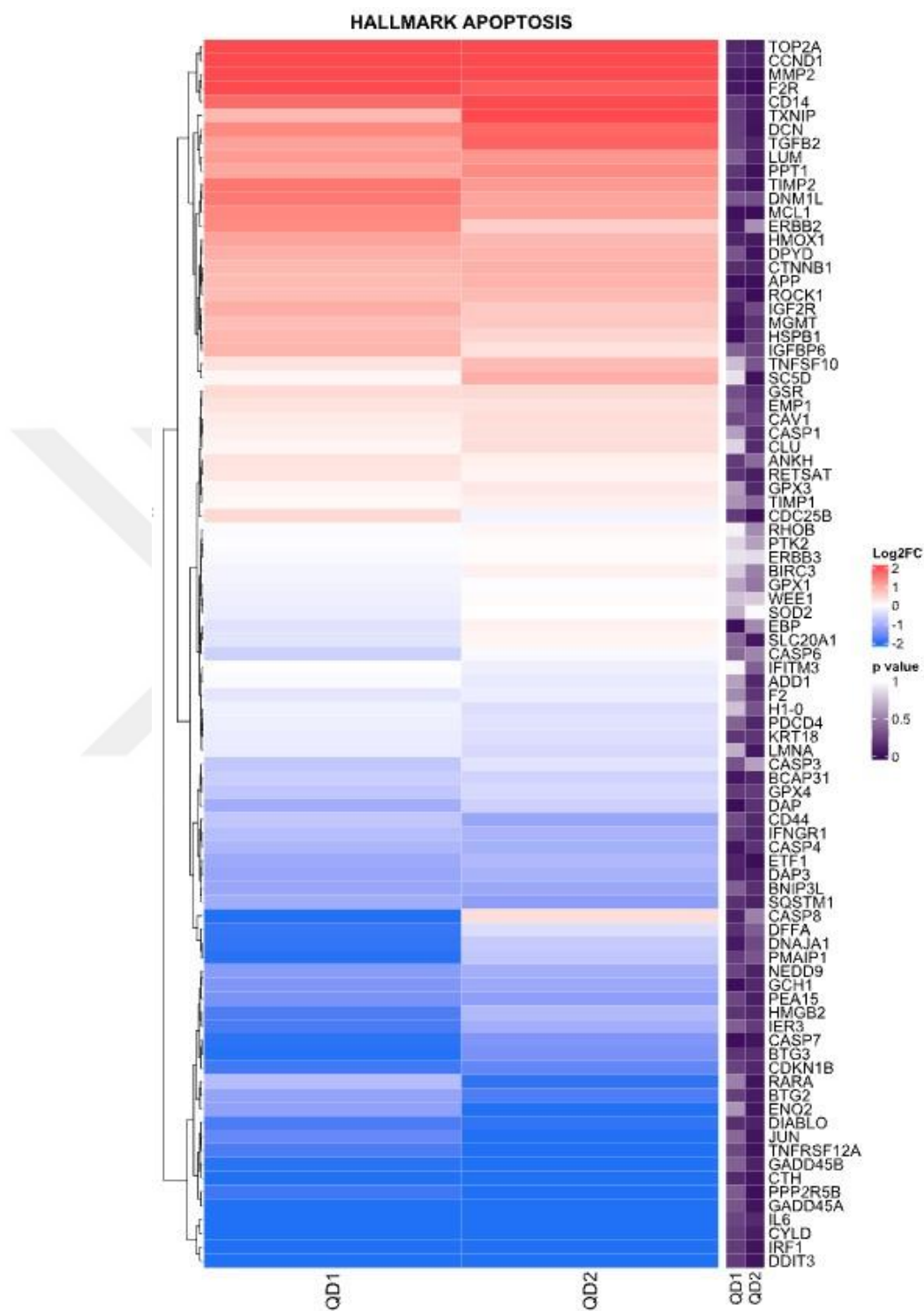


Figure 27: Heatmap of the apoptotic genes for quantum nanoparticles.

3.4.2.5. Metastasis and Tumorigenic Genes

Even though nanoparticles are used for the cancer treatment, there is a potential risk that they might cause cancer themselves by causing DNA damage. [99] Nanoparticle treated cells showed expression of metastasis genes. (Figure 28) ANLN gene that is deregulated almost in all NPs causes cellular migration of cancer cells and tumor progression. [100]

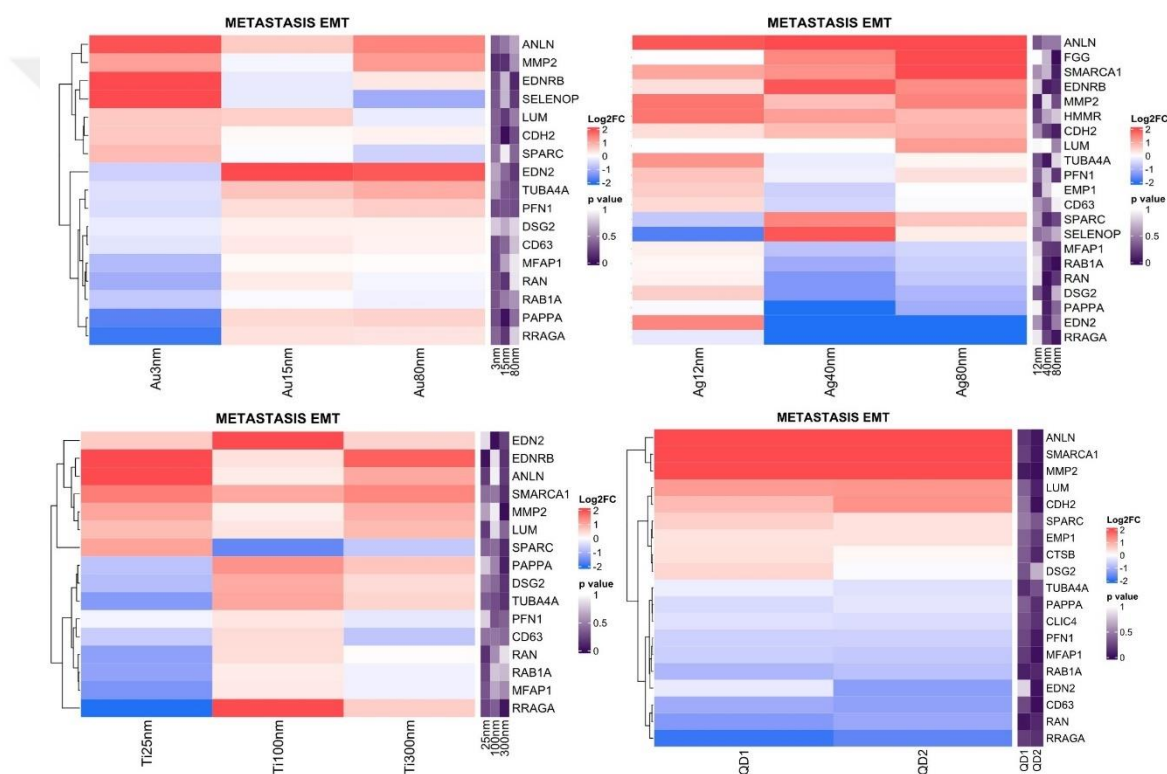


Figure 28: Heatmaps of the metastasis genes in NP treated cells.

Deregulation of liver cancer related genes were also noted. (Figure 29) According to the heatmaps, we see that there is not a certain pattern between different materials and sizes of the nanoparticles. Deregulation of the genes differs in between the nanoparticles which means that their toxicity mechanisms depend on the material and size of the NP.

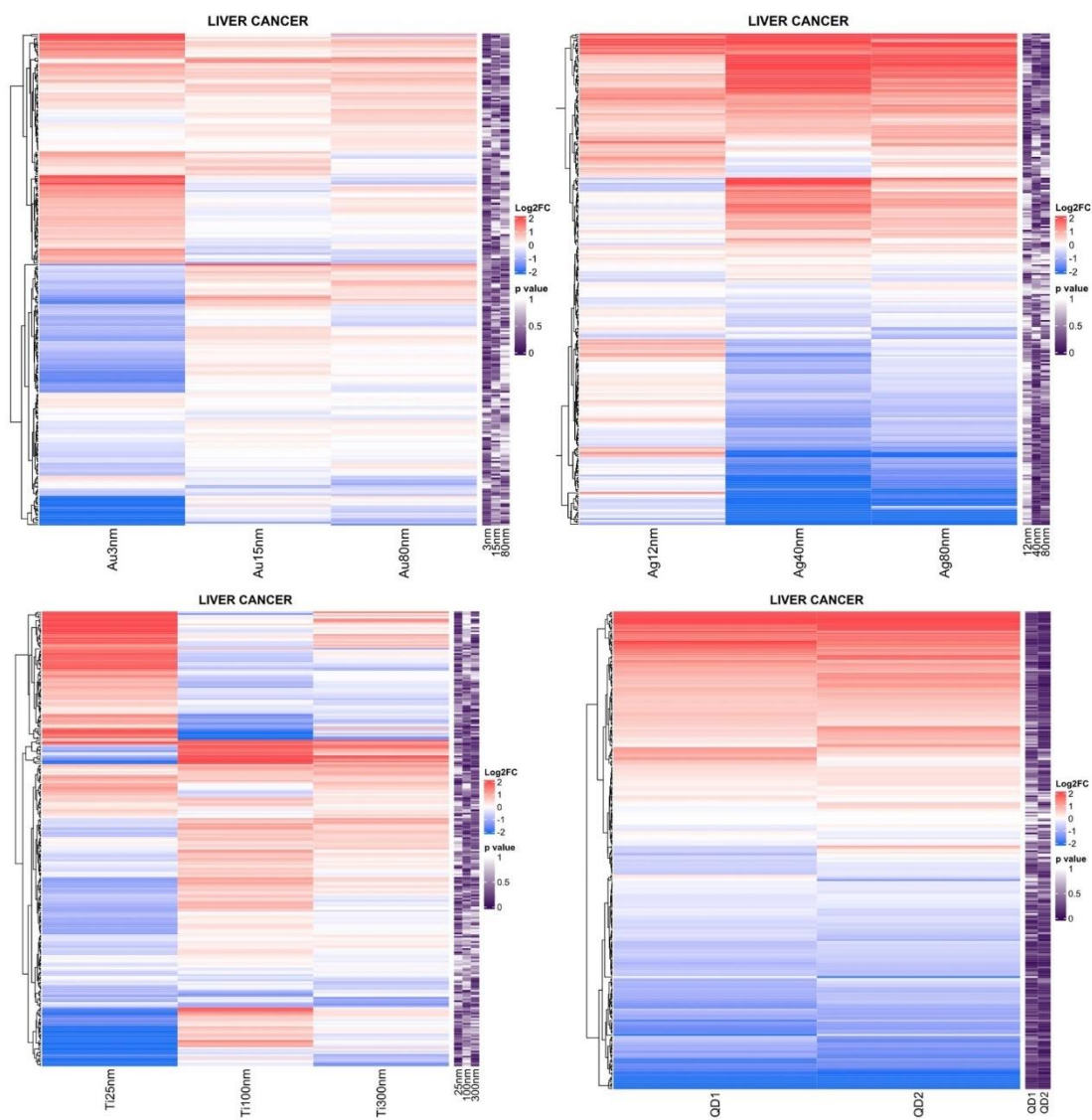


Figure 29: Heatmap of the genes related to liver cancer.

3.5. Discussion

Transcriptomic analysis enables us to see the effect of a stimuli at the gene expression level. Here in this study, we used eleven different nanoparticles of differing materials and sizes to monitor the toxicity of these materials on HepaRG cells. HepaRG cells are used for having the most similar drug metabolizing enzymatic activity to the primary hepatocytes. These cells are cultured in 3D spheroids so that it will be the most similar environment to the liver organ with actual cell-cell interactions and microcellular environment.

Nanoparticle toxicity is a newly arising concern since the applications of them is increasing every day. We use sunscreens that contain NPs, we eat food that contain NPs, and we breath the air that contain NPs. High exposure of human body to the NPs makes the further evaluation of the biosafety of the NPs essential. However, evaluation of the nanoparticle toxicity with viability assays stays limited since the nanoparticles interfere with the measurements. Also, studies that evaluated NP toxicity until now stays insufficient in terms of comparison of different materials and sizes of the nanoparticles as well as the techniques and cell line that they use.

Generally, toxicity assays are done with tumorigenic cell lines that does not have similar enzymatic activity as the primary hepatocytes. [103][104] That is why the results that they get does not reflect the real responses of the hepatocytes. For overcoming this obstacle, we are using HepaRG cells. Also culturing technique is also very important because cell-cell junctions and microenvironment effects how the cells respond to NP toxicity. [105][106][107] That is why we decided to use 3D spheroids so the cells will have cell-cell interactions and also have a communication through cellular microenvironment.

However, in this project we could not check the internalization of NPs into the spheroids. We might have sliced the spheroids after NP treatment, then we might check if the NPs can diffuse into the spheroids by looking them at the confocal microscopy.

The viability assessment analysis of the cells after NP exposure was checked with MTT assay. The results show the cell viability after NP toxicity, however, we see that the viability of the cells did not decrease with the increasing size of the NP which was expected. This might be because the small NPs can pass through the cell membrane easily, however big NPs cannot pass through the cell membrane, but they interact with the cell membrane and cause its disruption. Also, NPs when added into the cell medium, might adsorb some molecules on its surface. They might adsorb proteins, lipids and other molecules which is called protein corona. Formation of protein corona on NPs affects their cellular uptake, interaction with the cells, ROS production, and their cytotoxicity. Protein corona formation might cause some changes in the NPs such as it might change their size and surface charge which later change their toxicity potential and their interactions with the cells.

Transcriptomic analysis of the nanoparticle toxicity was essential for the evaluation of the nanoparticles in terms of their biosafety. When we have a comprehensive knowledge about how each type of nanoparticle interacts with cells and how they cause toxicity, we will be able to prevent this toxicity by using correct type of nanoparticle or modify the toxic ones to make them less harmful to cells.

RNA-seq analysis involves sequencing of total mRNA of the cells. This enables us to track the changes in the gene expressions in the cells. Cells change their expression profiles in order to defend the cell from foreign substances, get rid of the xenobiotics and survive the hard conditions. [106] In the nanoparticle exposed cells we see that there are

changes in the gene expression profiles. Because nanoparticles are shown to cause cytotoxicity on cells. [107] This shows how the nanoparticles cause toxicity and how the cells started expressions of stress response and survival genes. HepaRG cells changed the expression of many genes after NP exposure. There are 5 genes shared among all NPs which are all upregulated. These genes that are shared among all materials of nanoparticles are, OPN3, HS3ST3B1, AC078850.1, NUP85, IMPA2. The biological functions of these genes are associated to tumorigenesis, apoptosis, inflammation, liver disease and metastasis. This means nanoparticle treatment resulted in significant cellular damage and carcinogenic effect on HepaRG spheroids.

Hallmark gene set analysis showed that the genes that were significantly deregulated were enriched in the biological processes such as cell death, estrogen response, iron metabolism, metastasis, hypoxia, p53 pathway and peroxisome. Apoptosis and p53 pathways were thought to be deregulated because of the negative impact of the NPs to the cells. Apoptotic genes such as CD14, APP, TOP2A, DCN, IGFBP6, TIMP2, MMP2, F2R, CCND1 were found to be deregulated some of them were upregulated and some of them were downregulated. Differential expression of GCLM, HBB and ENDOD1 genes enriched the metal ion response and metal transporter gene sets. Metal transporters functions in the entrance of the NPs into the cells and metal ion responsive genes are the detox enzymes that metabolize these particles. Metal ion responsive genes probably enriched the heme metabolism ontology. Nanoparticles dysregulate the endocrine system and cause ROS production. [108] That might show the reason of the enrichment of the estrogen response. It is known that reactive oxygen species triggers coagulation. [109] Peroxisome ontology was enriched by the elevated ROS levels in the cells. Gene ontology

analysis revealed that DNA synthesis related genes were differentially expressed which in the nanoparticle treated cells. This may show that there is DNA damage response in the cells which might be because of the nanoparticle toxicity and reactive oxygen species. Nanoparticles are able to cause damage in the cell membrane. [110] This might explain the overexpression of genes that are linked to lipid metabolism and membrane depolarization. Nanoparticles release metal ions. [113][114] That means when the cells are exposed with nanoparticles, they are also exposed to metal ion that they are made of. Excessive metal ions might cause imbalance in the cation content of the cells, leading to cellular toxicity. So, the overexpression of genes linked to metal ion response is not unforeseen and shows that cells are trying to detoxify these metal ions.

Nanoparticles have strong oxidation capacity and they cause rapid production of ROS in the treated cells. [64] Rapid production of ROS causes DNA damage, organelles and cell membrane disruption, also cause lipid peroxidation and cell death. [113] When ROS production increases in the cells, detoxification pathways starts to be expressed in the cells. [114] Overexpression of antioxidant genes TXN, ATOX1, PTPA are because of the high levels of reactive oxygen species. Increased levels of ROS are closely linked to inflammation. TNF-alpha and interleukin pathway enrichments show inflammation in cells in response to nanoparticle toxicity. In Figure 23, we see that different nanoparticles have activated different genes and they have different pattern in their heatmaps. This might be because of production of different molecules of ROS (such as superoxide (O_2^-), hydroxyl radical ($\cdot OH$)). Since NPs have different surface charge, they might have different effect on the cells and might produce different ROS molecules. This can be determined by checking the genes that are activated by each NP. If a gene is activated by

superoxide radical, they the specific NP might be judged to produce this radical specifically.

Nanoparticles cause DNA damage, oxidative stress, inflammation that is because of their nano-size which helps them pass through the cell membrane easily. [115] Nanoparticles also have genotoxic effects which is the proof of the carcinogenic potential of the nanoparticles [118] [119] [120] By causing DNA damage, nanoparticles have the capacity of tumorigenesis that induces normal cells to evolve into cancer cells. In our results, we found that NP treatment cause deregulation of epithelial to mesenchymal transition genes (MMP2, ITGB5, MMP14,TIMP1,) and the genes that are associated to liver cancer. Expression of pathways associated to several types of cancer indicates the carcinogenic effects of NPs.

Oxidative stress, cell death, metastasis and tumorigenesis are the main results of nanoparticle cytotoxicity. Physicochemical properties of NPs impact their interaction with the cells and other compounds and their toxicity capacity. Their size, surface, composition are the responsible factors for their distinct behaviors and effects. Interaction between the nanoparticles and cells depend on these factors which results in the change of the toxicity mechanism of NPs. Different nanoparticles, because of having different physicochemical properties, have different impacts on the expression profiles of HepaRG cells. Even though, they show enrichment in similar pathways and oncology terms, they generally do this by deregulating distinct genes that is associated to the same biological process. This shows that even though the nanoparticles all exert a certain level of toxicity on HepaRG cells, the mechanism that they exert the stress is different.

3.6. Conclusion

In conclusion, nanoparticles are shown to induce stress on HepaRG spheroids. Transcriptomic analysis has revealed that nanoparticles cause excess production of reactive oxygen species, mitochondrial damage, DNA damage, inflammation, apoptosis and expression of cancerogenic and metastatic genes. We see that there are some common genes between the DEGs between different nanoparticles, in general the significantly deregulated genes are distinct for each nanoparticle type, we can see that from Venn diagrams of the common genes. The number of common DEGs for each nanoparticle is very low. This means that the mechanism of toxicity is different between nanoparticles even though they are consisting of the same material with differing sizes. Physicochemical properties of nanoparticles change their interaction with the cells and molecules, so they cause different expression profiles on cells.

CHAPTER 4

CONCLUSION AND FUTURE PERSPECTIVES

Nanoparticles are useful and effective structure that are used in a variety of fields including medicine, industry, manufacturing etc. Despite their advantages, they also have certain drawbacks. Cytotoxicity concern of nanoparticles has arisen, and many studies are focused on evaluating this toxicity. This study involves transcriptomic investigation of nanoparticle driven toxicity to enlighten the biosafety concerns about the nanoparticles. Findings of this article shows the level and mechanism of toxicity of different nanoparticles. Comprehensive knowledge about the mechanism of toxicity of each nanoparticle type enables us to define the needs and select the appropriate nanoparticle for certain applications. Also, this knowledge gives us possibility to prevent the adverse effects of these nanoparticles.

Human body exposure to these nanoparticles are increasing rapidly. We are all exposed to the nanoparticles from our food, the cosmetics that we use, from the medicine that we take when we are ill, even from the air that we breath. This high level of exposure necessitates an effective biosensor that will sense and signal for the presence of nanoparticles in certain media. A biosensor that was designed to sense the nanoparticle toxicity in the environment uses NFkB promoter as a sensing component. [119] This biosensor has a limitation that the promoter that they are using is a general stress responsive gene promoter. This means everything could activate the expression from this promoter. This means false positive results are inevitable. For a more sensitive biosensor

that will sense the toxic agent and give response in a dose dependent manner is what we need. In the future studies, we are aiming to use the results from this study and design an effective biosensor for detection of nanoparticles in the environment. Detection of the toxicity of the nanoparticles in a product is very important because now we lack that system to monitor the toxicity potential of nanoparticles in the products.

The biosensor that will be designed by using the promoter regions of the genes that responded to the nanoparticle toxicity will also be used for evaluation of the toxicity of other xenobiotics such as drugs. Drug toxicity will be detected by this biosensor in the preclinical stages of the new drugs to have a preliminary toxicity data about the drug before getting into in vivo experiments. With the help of this system maybe we could decrease the need for animal experiments which is unethical.

BIBLIOGRAPHY

- [1] A. F. Burlec *et al.*, “Current Overview of Metal Nanoparticles’ Synthesis, Characterization, and Biomedical Applications, with a Focus on Silver and Gold Nanoparticles,” *Pharmaceuticals*, vol. 16, no. 10, Art. no. 10, Oct. 2023, doi: 10.3390/ph16101410.
- [2] G. Bystrzejewska-Piotrowska, J. Golimowski, and P. L. Urban, “Nanoparticles: Their potential toxicity, waste and environmental management,” *Waste Manag.*, vol. 29, no. 9, pp. 2587–2595, Sep. 2009, doi: 10.1016/j.wasman.2009.04.001.
- [3] Y. Khan *et al.*, “Classification, Synthetic, and Characterization Approaches to Nanoparticles, and Their Applications in Various Fields of Nanotechnology: A Review,” *Catalysts*, vol. 12, no. 11, p. 1386, Nov. 2022, doi: 10.3390/catal12111386.
- [4] S. S. Patil and D. S. A. Nitave, “A REVIEW ON THE NANOPARTICLES,” *World J. Pharm. Res.*.
- [5] S. A. M. Ealia and M. P. Saravanakumar, “A review on the classification, characterisation, synthesis of nanoparticles and their application,” *IOP Conf. Ser. Mater. Sci. Eng.*, vol. 263, no. 3, p. 032019, Nov. 2017, doi: 10.1088/1757-899X/263/3/032019.
- [6] N. Joudeh and D. Linke, “Nanoparticle classification, physicochemical properties, characterization, and applications: a comprehensive review for biologists,” *J. Nanobiotechnology*, vol. 20, no. 1, p. 262, Jun. 2022, doi: 10.1186/s12951-022-01477-8.
- [7] D. Singh, S. Singh, J. Sahu, S. Srivastava, and M. R. Singh, “Ceramic nanoparticles: Recompense, cellular uptake and toxicity concerns,” *Artif. Cells Nanomedicine Biotechnol.*, vol. 44, no. 1, pp. 401–409, Jan. 2016, doi: 10.3109/21691401.2014.955106.
- [8] M. De, P. S. Ghosh, and V. M. Rotello, “Applications of Nanoparticles in Biology,” *Adv. Mater.*, vol. 20, no. 22, pp. 4225–4241, 2008, doi: 10.1002/adma.200703183.
- [9] A. López-Serrano, R. M. Olivas, J. S. Landaluze, and C. Cámara, “Nanoparticles: a global vision. Characterization, separation, and quantification methods. Potential environmental and health impact,” *Anal. Methods*, vol. 6, no. 1, pp. 38–56, Dec. 2013, doi: 10.1039/C3AY40517F.
- [10] I. Khan, K. Saeed, and I. Khan, “Nanoparticles: Properties, applications and toxicities,” *Arab. J. Chem.*, vol. 12, no. 7, pp. 908–931, Nov. 2019, doi: 10.1016/j.arabjc.2017.05.011.
- [11] E. Roduner, “Size matters: why nanomaterials are different,” *Chem. Soc. Rev.*, vol. 35, no. 7, pp. 583–592, Jun. 2006, doi: 10.1039/B502142C.
- [12] X. Bai *et al.*, “The Basic Properties of Gold Nanoparticles and their Applications in Tumor Diagnosis and Treatment,” *Int. J. Mol. Sci.*, vol. 21, no. 7, Art. no. 7, Jan. 2020, doi: 10.3390/ijms21072480.
- [13] M. M. Ghobashy, Sh. A. Alkhursani, H. A. Alqahtani, T. K. El-damhougy, and M. Madani, “Gold nanoparticles in microelectronics advancements and biomedical applications,” *Mater. Sci. Eng. B*, vol. 301, p. 117191, Mar. 2024, doi: 10.1016/j.mseb.2024.117191.

- [14] S. J. Cameron, F. Hosseini, and W. G. Willmore, "A Current Overview of the Biological and Cellular Effects of Nanosilver," *Int. J. Mol. Sci.*, vol. 19, no. 7, Art. no. 7, Jul. 2018, doi: 10.3390/ijms19072030.
- [15] X.-F. Zhang, Z.-G. Liu, W. Shen, and S. Gurunathan, "Silver Nanoparticles: Synthesis, Characterization, Properties, Applications, and Therapeutic Approaches," *Int. J. Mol. Sci.*, vol. 17, no. 9, p. 1534, Sep. 2016, doi: 10.3390/ijms17091534.
- [16] A. Kedziora, K. Gorzelańczyk, and G. Bugla-Płoskońska, "Positive and negative aspects of silver nanoparticles usage," *Biol Int*, vol. 53, pp. 67–76, Jan. 2013.
- [17] A. Ravindran, P. Chandran, and S. S. Khan, "Biofunctionalized silver nanoparticles: Advances and prospects," *Colloids Surf. B Biointerfaces*, vol. 105, pp. 342–352, May 2013, doi: 10.1016/j.colsurfb.2012.07.036.
- [18] "Titanium Dioxide Nanomaterials: Synthesis, Properties, Modifications, and Applications | Chemical Reviews." Accessed: Aug. 29, 2024. [Online]. Available: <https://pubs.acs.org/doi/10.1021/cr0500535>
- [19] A.-I. Moreno-Vega, T. Gómez-Quintero, R.-E. Nuñez-Anita, L.-S. Acosta-Torres, and V. Castaño, "Polymeric and Ceramic Nanoparticles in Biomedical Applications," *J. Nanotechnol.*, vol. 2012, no. 1, p. 936041, 2012, doi: 10.1155/2012/936041.
- [20] A. D. Racovita, "Titanium Dioxide: Structure, Impact, and Toxicity," *Int. J. Environ. Res. Public Health*, vol. 19, no. 9, p. 5681, May 2022, doi: 10.3390/ijerph19095681.
- [21] T. Ayorinde and C. M. Sayes, "An updated review of industrially relevant titanium dioxide and its environmental health effects," *J. Hazard. Mater. Lett.*, vol. 4, p. 100085, Nov. 2023, doi: 10.1016/j.hazl.2023.100085.
- [22] S. Çeşmeli and C. Biray Avci, "Application of titanium dioxide (TiO₂) nanoparticles in cancer therapies," *J. Drug Target.*, vol. 27, no. 7, pp. 762–766, Aug. 2019, doi: 10.1080/1061186X.2018.1527338.
- [23] J. Drbohlavova, V. Adam, R. Kizek, and J. Hubalek, "Quantum Dots — Characterization, Preparation and Usage in Biological Systems," *Int. J. Mol. Sci.*, vol. 10, no. 2, pp. 656–673, Feb. 2009, doi: 10.3390/ijms10020656.
- [24] K. McNamara and S. A. M. Tofail, "Nanoparticles in biomedical applications," *Adv. Phys. X*, vol. 2, no. 1, pp. 54–88, Jan. 2017, doi: 10.1080/23746149.2016.1254570.
- [25] M. Arruebo, R. Fernández-Pacheco, M. R. Ibarra, and J. Santamaría, "Magnetic nanoparticles for drug delivery," *Nano Today*, vol. 2, no. 3, pp. 22–32, Jun. 2007, doi: 10.1016/S1748-0132(07)70084-1.
- [26] E. Samaridou, J. Heyes, and P. Lutwyche, "Lipid nanoparticles for nucleic acid delivery: Current perspectives," *Adv. Drug Deliv. Rev.*, vol. 154–155, pp. 37–63, Jan. 2020, doi: 10.1016/j.addr.2020.06.002.
- [27] S. Mourdikoudis, R. M. Pallares, and N. T. K. Thanh, "Characterization techniques for nanoparticles: comparison and complementarity upon studying nanoparticle properties," *Nanoscale*, vol. 10, no. 27, pp. 12871–12934, Jul. 2018, doi: 10.1039/C8NR02278J.
- [28] D. N. G. Krishna and J. Philip, "Review on surface-characterization applications of X-ray photoelectron spectroscopy (XPS): Recent developments and challenges,"

- Appl. Surf. Sci. Adv.*, vol. 12, p. 100332, Dec. 2022, doi: 10.1016/j.apsadv.2022.100332.
- [29] “Biocompatible green-synthesized nanomaterials for therapeutic applications - ScienceDirect.” Accessed: Sep. 03, 2024. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/B9780323952538000127?via%3Dihub>
- [30] “Robust Nanoparticle Morphology and Size Analysis by Atomic Force Microscopy for Standardization.” Accessed: Sep. 03, 2024. [Online]. Available: https://www.jstage.jst.go.jp/article/cpb/68/8/68_c20-00311/_article
- [31] A. Rao *et al.*, “Characterization of nanoparticles using Atomic Force Microscopy,” *J. Phys. Conf. Ser.*, vol. 61, no. 1, p. 971, Apr. 2007, doi: 10.1088/1742-6596/61/1/192.
- [32] M. M. Eid, “Characterization of Nanoparticles by FTIR and FTIR-Microscopy,” in *Handbook of Consumer Nanoproducts*, S. Mallakpour and C. M. Hussain, Eds., Singapore: Springer Nature, 2022, pp. 645–673. doi: 10.1007/978-981-16-8698-6_89.
- [33] J. Stetefeld, S. A. McKenna, and T. R. Patel, “Dynamic light scattering: a practical guide and applications in biomedical sciences,” *Biophys. Rev.*, vol. 8, no. 4, pp. 409–427, Dec. 2016, doi: 10.1007/s12551-016-0218-6.
- [34] G. George, R. Wilson, and J. Joy, “Chapter 3 - Ultraviolet Spectroscopy: A Facile Approach for the Characterization of Nanomaterials,” in *Spectroscopic Methods for Nanomaterials Characterization*, S. Thomas, R. Thomas, A. K. Zachariah, and R. K. Mishra, Eds., in *Micro and Nano Technologies*, Elsevier, 2017, pp. 55–72. doi: 10.1016/B978-0-323-46140-5.00003-0.
- [35] L. Sun *et al.*, “Smart nanoparticles for cancer therapy,” *Signal Transduct. Target. Ther.*, vol. 8, no. 1, pp. 1–28, Nov. 2023, doi: 10.1038/s41392-023-01642-x.
- [36] M. Murakami *et al.*, “Improving drug potency and efficacy by nanocarrier-mediated subcellular targeting,” *Sci. Transl. Med.*, vol. 3, no. 64, p. 64ra2, Jan. 2011, doi: 10.1126/scitranslmed.3001385.
- [37] M. E. R. O’Brien *et al.*, “Reduced cardiotoxicity and comparable efficacy in a phase III trial of pegylated liposomal doxorubicin HCl (CAELYX/Doxil) versus conventional doxorubicin for first-line treatment of metastatic breast cancer,” *Ann. Oncol. Off. J. Eur. Soc. Med. Oncol.*, vol. 15, no. 3, pp. 440–449, Mar. 2004, doi: 10.1093/annonc/mdh097.
- [38] W. Li *et al.*, “Overcoming ABC transporter-mediated multidrug resistance: Molecular mechanisms and novel therapeutic drug strategies,” *Drug Resist. Updat. Rev. Comment. Antimicrob. Anticancer Chemother.*, vol. 27, pp. 14–29, Jul. 2016, doi: 10.1016/j.drug.2016.05.001.
- [39] Y. Li *et al.*, “Recent Advances in Mesoporous Silica Nanoparticle-Based Targeted Drug-Delivery Systems for Cancer Therapy,” *Nanomed.*, vol. 17, no. 18, pp. 1253–1279, Aug. 2022, doi: 10.2217/nnm-2022-0023.
- [40] M. J. Mitchell, M. M. Billingsley, R. M. Haley, M. E. Wechsler, N. A. Peppas, and R. Langer, “Engineering precision nanoparticles for drug delivery,” *Nat. Rev. Drug Discov.*, vol. 20, no. 2, pp. 101–124, Feb. 2021, doi: 10.1038/s41573-020-0090-8.
- [41] G. Fytianos, A. Rahdar, and G. Z. Kyzas, “Nanomaterials in Cosmetics: Recent Updates,” *Nanomaterials*, vol. 10, no. 5, p. 979, May 2020, doi: 10.3390/nano10050979.

- [42] P. Ganguli and S. Chaudhuri, "Nanomaterials in antimicrobial paints and coatings to prevent biodegradation of man-made surfaces: A review," *Mater. Today Proc.*, vol. 45, pp. 3769–3777, Jan. 2021, doi: 10.1016/j.matpr.2021.01.275.
- [43] A. P. S. Sawhney, B. Condon, K. V. Singh, S. S. Pang, G. Li, and D. Hui, "Modern Applications of Nanotechnology in Textiles," *Text. Res. J.*, vol. 78, no. 8, pp. 731–739, Aug. 2008, doi: 10.1177/0040517508091066.
- [44] B. Smolkova, N. El Yamani, A. R. Collins, A. C. Gutleb, and M. Dusinska, "Nanoparticles in food. Epigenetic changes induced by nanomaterials and possible impact on health," *Food Chem. Toxicol.*, vol. 77, pp. 64–73, Mar. 2015, doi: 10.1016/j.fct.2014.12.015.
- [45] O. of the Commissioner, "Considering Whether an FDA-Regulated Product Involves the Application of Nanotechnology," *U.S. Food and Drug Administration*. Apr. 2019. Accessed: May 26, 2023. [Online]. Available: <https://www.fda.gov/regulatory-information/search-fda-guidance-documents/considering-whether-fda-regulated-product-involves-application-nanotechnology>
- [46] Y. Ma, Z. Cong, P. Gao, and Y. Wang, "Nanosuspensions technology as a master key for nature products drug delivery and *In vivo* fate," *Eur. J. Pharm. Sci.*, vol. 185, p. 106425, Jun. 2023, doi: 10.1016/j.ejps.2023.106425.
- [47] A. Onoda, K. Takeda, and M. Umezawa, "Dose-dependent induction of astrocyte activation and reactive astrogliosis in mouse brain following maternal exposure to carbon black nanoparticle," *Part. Fibre Toxicol.*, vol. 14, no. 1, p. 4, Feb. 2017, doi: 10.1186/s12989-017-0184-6.
- [48] H. Zhang *et al.*, "Air pollution nanoparticle exposure reduces neurotrophin signaling and causes hippocampal neural stem cell quiescence in mouse brain," *Alzheimers Dement.*, vol. 18, no. S10, p. e063451, 2022, doi: 10.1002/alz.063451.
- [49] G. Martínez *et al.*, "Environmental Impact of Nanoparticles' Application as an Emerging Technology: A Review," *Materials*, vol. 14, no. 1, p. 166, Dec. 2020, doi: 10.3390/ma14010166.
- [50] "Phagocytosis - an overview | ScienceDirect Topics." Accessed: Aug. 31, 2024. [Online]. Available: <https://www.sciencedirect.com/topics/neuroscience/phagocytosis>
- [51] "Cellular Uptake, Intracellular Trafficking, and Cytotoxicity of Nanomaterials - Zhao - 2011 - Small - Wiley Online Library." Accessed: Aug. 31, 2024. [Online]. Available: <https://onlinelibrary.wiley.com/doi/10.1002/sml.201100001>
- [52] G. Sahay, D. Y. Alakhova, and A. V. Kabanov, "Endocytosis of nanomedicines," *J. Controlled Release*, vol. 145, no. 3, pp. 182–195, Aug. 2010, doi: 10.1016/j.jconrel.2010.01.036.
- [53] G. Patel, C. Patra, S. P. Srinivas, M. Kumawat, P. N. Navya, and H. K. Daima, "Methods to evaluate the toxicity of engineered nanomaterials for biomedical applications: a review," *Environ. Chem. Lett.*, vol. 19, no. 6, pp. 4253–4274, Dec. 2021, doi: 10.1007/s10311-021-01280-1.
- [54] "Nanoparticle Size Effects in Biomedical Applications | ACS Applied Nano Materials." Accessed: Sep. 01, 2024. [Online]. Available: <https://pubs.acs.org/doi/10.1021/acsanm.1c00987>

- [55] M. C. Arno *et al.*, “Exploiting the role of nanoparticle shape in enhancing hydrogel adhesive and mechanical properties,” *Nat. Commun.*, vol. 11, no. 1, p. 1420, Mar. 2020, doi: 10.1038/s41467-020-15206-y.
- [56] “Polymers and inorganic nanoparticles: A winning combination towards assembled nanostructures for cancer imaging and therapy - ScienceDirect.” Accessed: Sep. 01, 2024. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S1748013220302164?via%3Dihub>
- [57] “Quantitative Determination of the Hydrophobicity of Nanoparticles | Analytical Chemistry.” Accessed: Sep. 01, 2024. [Online]. Available: <https://pubs.acs.org/doi/10.1021/acs.analchem.1c04172>
- [58] “Surface functionalization of nanoparticles for nanomedicine - Chemical Society Reviews (RSC Publishing).” Accessed: Sep. 01, 2024. [Online]. Available: <https://pubs.rsc.org/en/content/articlelanding/2012/cs/c2cs15294k>
- [59] B. Saltepe, E. U. Bozkurt, N. Hacıosmanoğlu, and U. Ö. Ş. Şeker, “Genetic Circuits To Detect Nanomaterial Triggered Toxicity through Engineered Heat Shock Response Mechanism,” *ACS Synth. Biol.*, vol. 8, no. 10, pp. 2404–2417, Oct. 2019, doi: 10.1021/acssynbio.9b00291.
- [60] A. Seaton, L. Tran, R. Aitken, and K. Donaldson, “Nanoparticles, human health hazard and regulation,” *J. R. Soc. Interface*, vol. 7 Suppl 1, no. Suppl 1, pp. S119–129, Feb. 2010, doi: 10.1098/rsif.2009.0252.focus.
- [61] L. M. Brown *et al.*, “Ultrafine particles: mechanisms of lung injury,” *Philos. Trans. R. Soc. Lond. Ser. Math. Phys. Eng. Sci.*, vol. 358, no. 1775, pp. 2741–2749, Oct. 2000, doi: 10.1098/rsta.2000.0681.
- [62] A. B. Sengul and E. Asmatulu, “Toxicity of metal and metal oxide nanoparticles: a review,” *Environ. Chem. Lett.*, vol. 18, no. 5, pp. 1659–1683, Sep. 2020, doi: 10.1007/s10311-020-01033-6.
- [63] “Full article: Role of oxidative stress in nanoparticles toxicity.” Accessed: Sep. 03, 2024. [Online]. Available: <https://www.tandfonline.com/doi/full/10.1080/10715762.2020.1859108>
- [64] A. Manke, L. Wang, and Y. Rojanasakul, “Mechanisms of Nanoparticle-Induced Oxidative Stress and Toxicity,” *BioMed Res. Int.*, vol. 2013, p. 942916, 2013, doi: 10.1155/2013/942916.
- [65] M. Horie and Y. Tabei, “Role of oxidative stress in nanoparticles toxicity,” *Free Radic. Res.*, vol. 55, no. 4, pp. 331–342, Apr. 2021, doi: 10.1080/10715762.2020.1859108.
- [66] R. K. Shukla, A. Badiye, K. Vajpayee, and N. Kapoor, “Genotoxic Potential of Nanoparticles: Structural and Functional Modifications in DNA,” *Front. Genet.*, vol. 12, Sep. 2021, doi: 10.3389/fgene.2021.728250.
- [67] D. Wu, Y. Ma, Y. Cao, and T. Zhang, “Mitochondrial toxicity of nanomaterials,” *Sci. Total Environ.*, vol. 702, p. 134994, Feb. 2020, doi: 10.1016/j.scitotenv.2019.134994.
- [68] M. Summer *et al.*, “Inflammatory response of nanoparticles: Mechanisms, consequences, and strategies for mitigation,” *Chemosphere*, vol. 363, p. 142826, Sep. 2024, doi: 10.1016/j.chemosphere.2024.142826.
- [69] “The toxicity of nanoparticles and their interaction with cells: an in vitro metabolomic perspective - Nanoscale Advances (RSC Publishing).” Accessed:

- Sep. 03, 2024. [Online]. Available:
<https://pubs.rsc.org/en/content/articlelanding/2023/na/d2na00534d>
- [70] M. E. Tüçer *et al.*, “Transcriptomic investigation of NP toxicity on HepaRG spheroids,” *Chem. Biol. Interact.*, vol. 405, p. 111303, Jan. 2025, doi: 10.1016/j.cbi.2024.111303.
 - [71] S. Anthérieu, C. Chesné, R. Li, C. Guguen-Guillouzo, and A. Guillouzo, “Optimization of the HepaRG cell model for drug metabolism and toxicity studies,” *Toxicol. In Vitro*, vol. 26, no. 8, pp. 1278–1285, Dec. 2012, doi: 10.1016/j.tiv.2012.05.008.
 - [72] M.-J. Marion, O. Hantz, and D. Durantel, “The HepaRG Cell Line: Biological Properties and Relevance as a Tool for Cell Biology, Drug Metabolism, and Virology Studies,” in *Hepatocytes: Methods and Protocols*, P. Maurel, Ed., in *Methods in Molecular Biology*, Totowa, NJ: Humana Press, 2010, pp. 261–272. doi: 10.1007/978-1-60761-688-7_13.
 - [73] S. Gunti, A. T. K. Hoke, K. P. Vu, and N. R. London, “Organoid and Spheroid Tumor Models: Techniques and Applications,” *Cancers*, vol. 13, no. 4, p. 874, Jan. 2021, doi: 10.3390/cancers13040874.
 - [74] H. Tseng *et al.*, “A spheroid toxicity assay using magnetic 3D bioprinting and real-time mobile device-based imaging,” *Sci. Rep.*, vol. 5, no. 1, p. 13987, Sep. 2015, doi: 10.1038/srep13987.
 - [75] E. Flampouri, S. Imar, K. OConnell, and B. Singh, “Spheroid-3D and Monolayer-2D Intestinal Electrochemical Biosensor for Toxicity/Viability Testing: Applications in Drug Screening, Food Safety, and Environmental Pollutant Analysis,” *ACS Sens.*, vol. 4, no. 3, pp. 660–669, Mar. 2019, doi: 10.1021/acssensors.8b01490.
 - [76] S. J. Fey and K. Wrzesinski, “Determination of Drug Toxicity Using 3D Spheroids Constructed From an Immortal Human Hepatocyte Cell Line,” *Toxicol. Sci.*, vol. 127, no. 2, pp. 403–411, Jun. 2012, doi: 10.1093/toxsci/kfs122.
 - [77] Y. Han, S. Gao, K. Muegge, W. Zhang, and B. Zhou, “Advanced Applications of RNA Sequencing and Challenges,” *Bioinforma. Biol. Insights*, vol. 9s1, p. BBI.S28991, Jan. 2015, doi: 10.4137/BBI.S28991.
 - [78] K. R. Kukurba and S. B. Montgomery, “RNA Sequencing and Analysis,” *Cold Spring Harb. Protoc.*, vol. 2015, no. 11, p. pdb.top084970, Nov. 2015, doi: 10.1101/pdb.top084970.
 - [79] P. Joseph, “Transcriptomics in toxicology,” *Food Chem. Toxicol. Int. J. Publ. Br. Ind. Biol. Res. Assoc.*, vol. 109, no. Pt 1, pp. 650–662, Nov. 2017, doi: 10.1016/j.fct.2017.07.031.
 - [80] S. M. Thomas, C. L. Ackert-Bicknell, M. J. Zuscik, and K. A. Payne, “Understanding the Transcriptomic Landscape to Drive New Innovations in Musculoskeletal Regenerative Medicine,” *Curr. Osteoporos. Rep.*, vol. 20, no. 2, pp. 141–152, Apr. 2022, doi: 10.1007/s11914-022-00726-x.
 - [81] R. M. Leggett, R. H. Ramirez-Gonzalez, B. J. Clavijo, D. Waite, and R. P. Davey, “Sequencing quality assessment tools to enable data-driven informatics for high throughput genomics,” *Front. Genet.*, vol. 4, p. 288, Dec. 2013, doi: 10.3389/fgene.2013.00288.

- [82] D. Kim, B. Langmead, and S. L. Salzberg, "HISAT: a fast spliced aligner with low memory requirements," *Nat. Methods*, vol. 12, no. 4, pp. 357–360, Apr. 2015, doi: 10.1038/nmeth.3317.
- [83] M. Pertea, G. M. Pertea, C. M. Antonescu, T.-C. Chang, J. T. Mendell, and S. L. Salzberg, "StringTie enables improved reconstruction of a transcriptome from RNA-seq reads," *Nat. Biotechnol.*, vol. 33, no. 3, pp. 290–295, Mar. 2015, doi: 10.1038/nbt.3122.
- [84] A. C. Frazee, G. Pertea, A. E. Jaffe, B. Langmead, S. L. Salzberg, and J. T. Leek, "Ballgown bridges the gap between transcriptome assembly and expression analysis," *Nat. Biotechnol.*, vol. 33, no. 3, p. 243, Mar. 2015, doi: 10.1038/nbt.3172.
- [85] M. Pertea, D. Kim, G. M. Pertea, J. T. Leek, and S. L. Salzberg, "Transcript-level expression analysis of RNA-seq experiments with HISAT, StringTie and Ballgown," *Nat. Protoc.*, vol. 11, no. 9, Art. no. 9, Sep. 2016, doi: 10.1038/nprot.2016.095.
- [86] M. Ashburner *et al.*, "Gene Ontology: tool for the unification of biology," *Nat. Genet.*, vol. 25, no. 1, pp. 25–29, May 2000, doi: 10.1038/75556.
- [87] K. Zhang *et al.*, "A novel function of IMPA2, plays a tumor-promoting role in cervical cancer," *Cell Death Dis.*, vol. 11, no. 5, pp. 1–14, May 2020, doi: 10.1038/s41419-020-2507-z.
- [88] X. Lei *et al.*, "IMPA2 promotes basal-like breast cancer aggressiveness by a MYC-mediated positive feedback loop," *Cancer Lett.*, vol. 582, p. 216527, Feb. 2024, doi: 10.1016/j.canlet.2023.216527.
- [89] Y. Tian, Q. Luo, K. Huang, T. Sun, and S. Luo, "Long Noncoding RNA AC078850.1 Induces NLRP3 Inflammasome-Mediated Pyroptosis in Atherosclerosis by Upregulating ITGB2 Transcription via Transcription Factor HIF-1 α ," *Biomedicines*, vol. 11, no. 6, p. 1734, Jun. 2023, doi: 10.3390/biomedicines11061734.
- [90] "Expression of OPN3 in lung adenocarcinoma promotes epithelial-mesenchymal transition and tumor metastasis - Xu - 2020 - Thoracic Cancer - Wiley Online Library." Accessed: Oct. 19, 2023. [Online]. Available: <https://onlinelibrary.wiley.com/doi/10.1111/1759-7714.13254>
- [91] Z. Zhang, H. Jiang, Y. Wang, and M. Shi, "Heparan sulfate D-glucosamine 3-O-sulfotransferase 3B1 is a novel regulator of transforming growth factor-beta-mediated epithelial-to-mesenchymal transition and regulated by miR-218 in nonsmall cell lung cancer," *J. Cancer Res. Ther.*, vol. 14, no. 1, pp. 24–29, Jan. 2018, doi: 10.4103/jcrt.JCRT_659_17.
- [92] K. Song, Q. Li, Z.-Z. Jiang, C.-W. Guo, and P. Li, "Heparan sulfate D-glucosaminyl 3-O-sulfotransferase-3B1, a novel epithelial-mesenchymal transition inducer in pancreatic cancer," *Cancer Biol. Ther.*, vol. 12, no. 5, pp. 388–398, Sep. 2011, doi: 10.4161/cbt.12.5.15957.
- [93] Y.-C. Wu *et al.*, "NUP85 alleviates lipid metabolism and inflammation by regulating PI3K/AKT signaling pathway in nonalcoholic fatty liver disease," *Int. J. Biol. Sci.*, vol. 20, no. 6, pp. 2219–2235, 2024, doi: 10.7150/ijbs.92337.
- [94] D. M. Grant, "Detoxification pathways in the liver," *J. Inherit. Metab. Dis.*, vol. 14, no. 4, pp. 421–430, 1991, doi: 10.1007/BF01797915.

- [95] B. Gilani and M. Cassagnol, "Biochemistry, Cytochrome P450," in *StatPearls*, Treasure Island (FL): StatPearls Publishing, 2024. Accessed: Sep. 04, 2024. [Online]. Available: <http://www.ncbi.nlm.nih.gov/books/NBK557698/>
- [96] L. Lei *et al.*, "A polymorphism in metallothionein 1A (*MT1A*) is associated with cadmium-related excretion of urinary beta 2-microglobulin," *Toxicol. Appl. Pharmacol.*, vol. 265, no. 3, pp. 373–379, Dec. 2012, doi: 10.1016/j.taap.2012.09.006.
- [97] H. L. Karlsson *et al.*, "Cell membrane damage and protein interaction induced by copper containing nanoparticles--importance of the metal release process," *Toxicology*, vol. 313, no. 1, pp. 59–69, Nov. 2013, doi: 10.1016/j.tox.2013.07.012.
- [98] P. Ruenraroengsak *et al.*, "Respiratory epithelial cytotoxicity and membrane damage (holes) caused by amine-modified nanoparticles," *Nanotoxicology*, vol. 6, no. 1, pp. 94–108, Feb. 2012, doi: 10.3109/17435390.2011.558643.
- [99] T. Prevenslik, "Nanoparticle toxicity and cancer," *J. Phys. Conf. Ser.*, vol. 304, no. 1, p. 012054, Jul. 2011, doi: 10.1088/1742-6596/304/1/012054.
- [100] K. Liu *et al.*, "Pan-cancer analysis of the prognostic and immunological role of ANLN: An onco-immunological biomarker," *Front. Genet.*, vol. 13, Aug. 2022, doi: 10.3389/fgene.2022.922472.
- [101] X. Jiang *et al.*, "Nanotoxicity of Silver Nanoparticles on HEK293T Cells: A Combined Study Using Biomechanical and Biological Techniques," *ACS Omega*, vol. 3, no. 6, pp. 6770–6778, Jun. 2018, doi: 10.1021/acsomega.8b00608.
- [102] R. V.g. and M. P.v., "Cellular interactions of zinc oxide nanoparticles with human embryonic kidney (HEK 293) cells," *Colloids Surf. B Biointerfaces*, vol. 157, pp. 182–190, Sep. 2017, doi: 10.1016/j.colsurfb.2017.05.069.
- [103] M. Ravi, V. Paramesh, S. R. Kaviya, E. Anuradha, and F. D. P. Solomon, "3D cell culture systems: advantages and applications," *J. Cell. Physiol.*, vol. 230, no. 1, pp. 16–26, Jan. 2015, doi: 10.1002/jcp.24683.
- [104] Y. Fang and R. Eglen, "Three-Dimensional Cell Cultures in Drug Discovery and Development," *SLAS Discov.*, p. 2472555217696795, Mar. 2017, doi: 10.1177/2472555217696795.
- [105] O. Habanjar, M. Diab-Assaf, F. Caldefie-Chezet, and L. Delort, "3D Cell Culture Systems: Tumor Application, Advantages, and Disadvantages," *Int. J. Mol. Sci.*, vol. 22, no. 22, p. 12200, Nov. 2021, doi: 10.3390/ijms222212200.
- [106] B. Alberts, A. Johnson, J. Lewis, M. Raff, K. Roberts, and P. Walter, "An Overview of Gene Control," in *Molecular Biology of the Cell. 4th edition*, Garland Science, 2002. Accessed: Sep. 05, 2024. [Online]. Available: <https://www.ncbi.nlm.nih.gov/books/NBK26885/>
- [107] P. Xiong *et al.*, "Cytotoxicity of Metal-Based Nanoparticles: From Mechanisms and Methods of Evaluation to Pathological Manifestations," *Adv. Sci.*, vol. 9, no. 16, p. 2106049, 2022, doi: 10.1002/advs.202106049.
- [108] R. Santacruz-Márquez, J. A. Flaws, L. del C. Sánchez-Peña, and I. Hernández-Ochoa, "Exposure to Zinc Oxide Nanoparticles Increases Estradiol Levels and Induces an Antioxidant Response in Antral Ovarian Follicles In Vitro," *Toxics*, vol. 11, no. 7, p. 602, Jul. 2023, doi: 10.3390/toxics11070602.
- [109] A. Bettiol *et al.*, "Erythrocyte oxidative stress and thrombosis," *Expert Rev. Mol. Med.*, vol. 24, p. e31, Aug. 2022, doi: 10.1017/erm.2022.25.

- [110] M. R. R. de Planque, S. Aghdaei, T. Roose, and H. Morgan, "Electrophysiological Characterization of Membrane Disruption by Nanoparticles," *ACS Nano*, vol. 5, no. 5, pp. 3599–3606, May 2011, doi: 10.1021/nn103320j.
- [111] "Multi-omics approaches confirm metal ions mediate the main toxicological pathways of metal-bearing nanoparticles in lung epithelial A549 cells - Environmental Science: Nano (RSC Publishing)." Accessed: Aug. 31, 2023. [Online]. Available: <https://pubs.rsc.org/en/content/articlelanding/2018/en/c8en00071a>
- [112] H. Ma *et al.*, "The Metal Ion Release of Manganese Ferrite Nanoparticles: Kinetics, Effects on Magnetic Resonance Relaxivities, and Toxicity," *ACS Appl. Bio Mater.*, vol. 5, no. 6, pp. 3067–3074, Jun. 2022, doi: 10.1021/acsabm.2c00338.
- [113] Z. Yu *et al.*, "Reactive Oxygen Species-Related Nanoparticle Toxicity in the Biomedical Field," *Nanoscale Res. Lett.*, vol. 15, no. 1, p. 115, Dec. 2020, doi: 10.1186/s11671-020-03344-7.
- [114] L. He, T. He, S. Farrar, L. Ji, T. Liu, and X. Ma, "Antioxidants Maintain Cellular Redox Homeostasis by Elimination of Reactive Oxygen Species," *Cell. Physiol. Biochem.*, vol. 44, no. 2, pp. 532–553, Nov. 2017, doi: 10.1159/000485089.
- [115] P. Chen, "Detection of DNA Damage Response Caused by Different Forms of Titanium Dioxide Nanoparticles using Sensor Cells," *J. Biosens. Bioelectron.*, vol. 03, no. 05, 2012, doi: 10.4172/2155-6210.1000129.
- [116] K. Klien and J. Godnić-Cvar, "Genotoxicity of metal nanoparticles: focus on in vivo studies," *Arh. Hig. Rada Toksikol.*, vol. 63, no. 2, pp. 133–145, Jun. 2012, doi: 10.2478/10004-1254-63-2012-2213.
- [117] M. Ahamed *et al.*, "DNA damage response to different surface chemistry of silver nanoparticles in mammalian cells," *Toxicol. Appl. Pharmacol.*, vol. 233, no. 3, pp. 404–410, Dec. 2008, doi: 10.1016/j.taap.2008.09.015.
- [118] H. K. Lim, P. V. AshaRani, and M. P. Hande, "Enhanced Genotoxicity of Silver Nanoparticles in DNA Repair Deficient Mammalian Cells," *Front. Genet.*, vol. 3, 2012, Accessed: Oct. 19, 2023. [Online]. Available: <https://www.frontiersin.org/articles/10.3389/fgene.2012.00104>
- [119] M. Dubiak-Szepietowska, A. Karczmarczyk, T. Winckler, and K.-H. Feller, "A cell-based biosensor for nanomaterials cytotoxicity assessment in three dimensional cell culture," *Toxicology*, vol. 370, pp. 60–69, Aug. 2016, doi: 10.1016/j.tox.2016.09.012.

APPENDIX A

Codes Used in The Study

[1] Build indexes for Hisat2

```
Hisat2-build -p 16 genome.fa genome
```

[2] Align the RNA-seq reads to the genome

```
hisat2 -threads 8 -dta -x indexes/genome -1 heparg1_1.fq.gz -2 heparg1_2.fq.gz -S  
heparg1.sam
```

```
hisat2 -threads 8 -dta -x indexes/genome -1 heparg2_1.fq.gz -2 heparg2_2.fq.gz -S  
heparg2.sam
```

```
hisat2 -threads 8 -dta -x indexes/genome -1 NP3nmAuNP1_1.fq.gz -2  
NP3nmAuNP1_2.fq.gz -S NP3nmAuNP1.sam
```

```
hisat2 -threads 8 -dta -x indexes/genome -1 NP3nmAuNP2_1.fq.gz -2  
NP3nmAuNP2_2.fq.gz -S NP3nmAuNP2.sam
```

```
hisat2 -threads 8 -dta -x indexes/genome -1 NP15nmAuNP1_1.fq.gz -2  
NP15nmAuNP1_2.fq.gz -S NP15nmAuNP1.sam
```

```
hisat2 -threads 8 -dta -x indexes/genome -1 NP15nmAuNP2_1.fq.gz -2  
NP15nmAuNP2_2.fq.gz -S NP15nmAuNP2.sam
```

```
hisat2 -threads 8 -dta -x indexes/genome -1 NP80nmAuNP1_1.fq.gz -2  
NP80nmAuNP1_2.fq.gz -S NP80nmAuNP1.sam
```

```
hisat2 -threads 8 -dta -x indexes/genome -1 NP80nmAuNP2_1.fq.gz -2  
NP80nmAuNP2_2.fq.gz -S NP80nmAuNP2.sam
```

```
hisat2 -threads 8 -dta -x indexes/genome -1 NP12nmAgNP1_1.fq.gz -2  
NP12nmAgNP1_2.fq.gz -S NP12nmAgNP1.sam
```

```
hisat2 -threads 8 -dta -x indexes/genome -1 NP12nmAgNP2_1.fq.gz -2  
NP12nmAgNP2_2.fq.gz -S NP12nmAgNP2.sam
```

```
hisat2 -threads 8 -dta -x indexes/genome -1 NP40nmAgNP1_1.fq.gz -2  
NP40nmAgNP1_2.fq.gz -S NP40nmAgNP1.sam
```

```

hisat2 -threads 8 -dta -x indexes/genome -1 NP40nmAgNP2_1.fq.gz -2
NP40nmAgNP2_2.fq.gz -S NP40nmAgNP2.sam
hisat2 -threads 8 -dta -x indexes/genome -1 NP80nmAgNP1_1.fq.gz -2
NP80nmAgNP1_2.fq.gz -S NP80nmAgNP1.sam
hisat2 -threads 8 -dta -x indexes/genome -1 NP80nmAgNP2_1.fq.gz -2
NP80nmAgNP2_2.fq.gz -S NP80nmAgNP2.sam
hisat2 -threads 8 -dta -x indexes/genome -1 NP25nmTiO2NP1_1.fq.gz -2
NP25nmTiO2NP1_2.fq.gz -S NP25nmTiO2NP1.sam
hisat2 -threads 8 -dta -x indexes/genome -1 NP25nmTiO2NP2_1.fq.gz -2
NP25nmTiO2NP2_2.fq.gz -S NP25nmTiO2NP2.sam
hisat2 -threads 8 -dta -x indexes/genome -1 NP100nmTiO2NP1_1.fq.gz -2
NP100nmTiO2NP1_2.fq.gz -S NP100nmTiO2NP1.sam
hisat2 -threads 8 -dta -x indexes/genome -1 NP100nmTiO2NP2_1.fq.gz -2
NP100nmTiO2NP2_2.fq.gz -S NP100nmTiO2NP2.sam
hisat2 -threads 8 -dta -x indexes/genome -1 NP300nmTiO2NP1_1.fq.gz -2
NP300nmTiO2NP1_2.fq.gz -S NP300nmTiO2NP1.sam
hisat2 -threads 8 -dta -x indexes/genome -1 NP300nmTiO2NP2_1.fq.gz -2
NP300nmTiO2NP2_2.fq.gz -S NP300nmTiO2NP2.sam
hisat2 -threads 8 -dta -x indexes/genome -1 NPQD1_1_1.fq.gz -2 NPQD1_1_2.fq.gz -S
NPQD1_1.sam
hisat2 -threads 8 -dta -x indexes/genome -1 NPQD1_2_1.fq.gz -2 NPQD1_2_2.fq.gz -S
NPQD2_2.sam
hisat2 -threads 8 -dta -x indexes/genome -1 NPQD2_1_1.fq.gz -2 NPQD2_1_2.fq.gz -S
NPQD2_1.sam
hisat2 -threads 8 -dta -x indexes/genome -1 NPQD2_2_1.fq.gz -2 NPQD2_2_2.fq.gz -S
NPQD2_2.sam

```

[3] Sort and convert the SAM files to BAM.

```

samtools sort -@ 8 -o heparg1.bam heparg1.sam
samtools sort -@ 8 -o heparg2.bam heparg2.sam
samtools sort -@ 8 -o NP3nmAuNP1.bam NP3nmAuNP1.sam
samtools sort -@ 8 -o NP3nmAuNP2.bam NP3nmAuNP2.sam

```

```

samtools sort -@ 8 -o NP15nmAuNP1.bam NP15nmAuNP1.sam
samtools sort -@ 8 -o NP15nmAuNP2.bam NP15nmAuNP2.sam
samtools sort -@ 8 -o NP80nmAuNP1.bam NP80nmAuNP1.sam
samtools sort -@ 8 -o NP80nmAuNP2.bam NP80nmAuNP2.sam
samtools sort -@ 8 -o NP12nmAgNP1.bam NP12nmAgNP1.sam
samtools sort -@ 8 -o NP12nmAgNP2.bam NP12nmAgNP2.sam
samtools sort -@ 8 -o NP40nmAgNP1.bam NP40nmAgNP1.sam
samtools sort -@ 8 -o NP40nmAgNP2.bam NP40nmAgNP2.sam
samtools sort -@ 8 -o NP80nmAgNP1.bam NP80nmAgNP1.sam
samtools sort -@ 8 -o NP80nmAgNP2.bam NP80nmAgNP2.sam
samtools sort -@ 8 -o NP25nmTiO2NP1.bam NP25nmTiO2NP1.sam
samtools sort -@ 8 -o NP25nmTiO2NP2.bam NP25nmTiO2NP2.sam
samtools sort -@ 8 -o NP100nmTiO2NP1.bam NP100nmTiO2NP1.sam
samtools sort -@ 8 -o NP100nmTiO2NP2.bam NP100nmTiO2NP2.sam
samtools sort -@ 8 -o NP300nmTiO2NP1.bam NP300nmTiO2NP1.sam
samtools sort -@ 8 -o NP300nmTiO2NP2.bam NP300nmTiO2NP2.sam
samtools sort -@ 8 -o NPQD1.bam NPQD1.sam
samtools sort -@ 8 -o NPQD2.bam NPQD2.sam

```

[4] Assemble transcripts for each sample.

```

stringtie -p 8 -G genes/genes.gtf -o heparg1.gtf -l heparg1.label heparg1.bam
stringtie -p 8 -G genes/genes.gtf -o heparg2.gtf -l heparg2.label heparg2.bam
stringtie -p 8 -G genes/genes.gtf -o NP3nmAuNP1.gtf -l NP3nmAuNP1.label
NP3nmAuNP1.bam
stringtie -p 8 -G genes/genes.gtf -o NP3nmAuNP2.gtf -l NP3nmAuNP2.label
NP3nmAuNP2.bam
stringtie -p 8 -G genes/genes.gtf -o NP15nmAuNP1.gtf -l NP15nmAuNP1.label
NP15nmAuNP1.bam
stringtie -p 8 -G genes/genes.gtf -o NP15nmAuNP2.gtf -l NP15nmAuNP2.label
NP15nmAuNP2.bam
stringtie -p 8 -G genes/genes.gtf -o NP80nmAuNP1.gtf -l NP80nmAuNP1.label
NP80nmAuNP1.bam

```

```

stringtie -p 8 -G genes/genes.gtf -o NP80nmAuNP2.gtf -l NP80nmAuNP2.label
NP80nmAuNP2.bam

stringtie -p 8 -G genes/genes.gtf -o NP12nmAgNP1.gtf -l NP12nmAgNP1.label
NP12nmAgNP1.bam

stringtie -p 8 -G genes/genes.gtf -o NP12nmAgNP2.gtf -l NP12nmAgNP2.label
NP12nmAgNP2.bam

stringtie -p 8 -G genes/genes.gtf -o NP40nmAgNP1.gtf -l NP40nmAgNP1.label
NP40nmAgNP1.bam

stringtie -p 8 -G genes/genes.gtf -o NP40nmAgNP2.gtf -l NP40nmAgNP2.label
NP40nmAgNP2.bam

stringtie -p 8 -G genes/genes.gtf -o NP80nmAgNP1.gtf -l NP80nmAgNP1.label
NP80nmAgNP1.bam

stringtie -p 8 -G genes/genes.gtf -o NP80nmAgNP2.gtf -l NP80nmAgNP2.label
NP80nmAgNP2.bam

stringtie -p 8 -G genes/genes.gtf -o NP25nmTiO2NP1.gtf -l NP25nmTiO2NP1.label
NP25nmTiO2NP1.bam

stringtie -p 8 -G genes/genes.gtf -o NP25nmTiO2NP2.gtf -l NP25nmTiO2NP2.label
NP25nmTiO2NP2.bam

stringtie -p 8 -G genes/genes.gtf -o NP100nmTiO2NP1.gtf -l NP100nmTiO2NP1.label
NP100nmTiO2NP1.bam

stringtie -p 8 -G genes/genes.gtf -o NP100nmTiO2NP2.gtf -l NP100nmTiO2NP2.label
NP100nmTiO2NP2.bam

stringtie -p 8 -G genes/genes.gtf -o NP300nmTiO2NP1.gtf -l NP300nmTiO2NP1.label
NP300nmTiO2NP1.bam

stringtie -p 8 -G genes/genes.gtf -o NP300nmTiO2NP2.gtf -l NP300nmTiO2NP2.label
NP300nmTiO2NP2.bam

stringtie -p 8 -G genes/genes.gtf -o NPQD1_1.gtf -l NPQD1_1.label NPQD1_1.bam
stringtie -p 8 -G genes/genes.gtf -o NPQD1_2.gtf -l NPQD1_2.label NPQD1_2.bam
stringtie -p 8 -G genes/genes.gtf -o NPQD2_1.gtf -l NPQD2_1.label NPQD2_1.bam
stringtie -p 8 -G genes/genes.gtf -o NPQD2_2.gtf -l NPQD2_2.label NPQD2_2.bam

```

[5] Merge transcripts from all samples.

Stringtie -merge -p 8 -G genes/genes.gtf -o stringtie_merged.gtf mergelist.txt

Mergelist.txt:

Heparg1.gtf
Heparg1.gtf
NP3nmAuNP1.gtf
NP3nmAuNP2.gtf
NP15nmAuNP1.gtf
NP15nmAuNP2.gtf
NP80nmAuNP1.gtf
NP80nmAuNP2.gtf
NP12nmAgNP1.gtf
NP12nmAgNP2.gtf
NP40nmAgNP1.gtf
NP40nmAgNP2.gtf
NP80nmAgNP1.gtf
NP80nmAgNP2.gtf
NP25nmTiO2NP1.gtf
NP25nmTiO2NP2.gtf
NP100nmTiO2NP1.gtf
NP100nmTiO2NP2.gtf
NP300nmTiO2NP1.gtf
NP300nmTiO2NP2.gtf
NPQD1_1.gtf
NPQD1_2.gtf
NPQD2_1.gtf
NPQD2_2.gtf

[6] Estimate transcript abundances and create table counts for Ballgown.

stringtie -e -B -p 8 -G stringtie_merged.gtf -o ballgown/heparg1/heparg1.gtf
heparg1.bam

```

stringtie -e -B -p 8 -G stringtie_merged.gtf -o ballgown/heparg2/heparg2.gtf
heparg2.bam
stringtie -e -B -p 8 -G stringtie_merged.gtf -o
ballgown/NP3nmAuNP1/NP3nmAuNP1.gtf NP3nmAuNP1.bam
stringtie -e -B -p 8 -G stringtie_merged.gtf -o
ballgown/NP3nmAuNP2/NP3nmAuNP2.gtf NP3nmAuNP2.bam
stringtie -e -B -p 8 -G stringtie_merged.gtf -o
ballgown/NP15nmAuNP1/NP15nmAuNP1.gtf NP15nmAuNP1.bam
stringtie -e -B -p 8 -G stringtie_merged.gtf -o
ballgown/NP15nmAuNP2/NP15nmAuNP2.gtf NP15nmAuNP2.bam
stringtie -e -B -p 8 -G stringtie_merged.gtf -o
ballgown/NP80nmAuNP1/NP80nmAuNP1.gtf NP80nmAuNP1.bam
stringtie -e -B -p 8 -G stringtie_merged.gtf -o
ballgown/NP80nmAuNP2/NP80nmAuNP2.gtf NP80nmAuNP2.bam
stringtie -e -B -p 8 -G stringtie_merged.gtf -o
ballgown/NP12nmAgNP1/NP12nmAgNP1.gtf NP12nmAgNP1.bam
stringtie -e -B -p 8 -G stringtie_merged.gtf -o
ballgown/NP12nmAgNP2/NP12nmAgNP2.gtf NP12nmAgNP2.bam
stringtie -e -B -p 8 -G stringtie_merged.gtf -o
ballgown/NP40nmAgNP1/NP40nmAgNP1.gtf NP40nmAgNP1.bam
stringtie -e -B -p 8 -G stringtie_merged.gtf -o
ballgown/NP40nmAgNP2/NP40nmAgNP2.gtf NP40nmAgNP2.bam
stringtie -e -B -p 8 -G stringtie_merged.gtf -o
ballgown/NP80nmAgNP1/NP80nmAgNP1.gtf NP80nmAgNP1.bam
stringtie -e -B -p 8 -G stringtie_merged.gtf -o
ballgown/NP80nmAgNP2/NP80nmAgNP2.gtf NP80nmAgNP2.bam
stringtie -e -B -p 8 -G stringtie_merged.gtf -o
ballgown/NP25nmTiO2NP1/NP25nmTiO2NP1.gtf NP25nmTiO2NP1.bam
stringtie -e -B -p 8 -G stringtie_merged.gtf -o
ballgown/NP25nmTiO2NP2/NP25nmTiO2NP2.gtf NP25nmTiO2NP2.bam

```

```

stringtie -e -B -p 8 -G stringtie_merged.gtf -o
ballgown/NP100nmTiO2NP1/NP100nmTiO2NP1.gtf NP100nmTiO2NP1.bam
stringtie -e -B -p 8 -G stringtie_merged.gtf -o
ballgown/NP100nmTiO2NP2/NP100nmTiO2NP2.gtf NP100nmTiO2NP2.bam
stringtie -e -B -p 8 -G stringtie_merged.gtf -o
ballgown/NP300nmTiO2NP1/NP300nmTiO2NP1.gtf NP300nmTiO2NP1.bam
stringtie -e -B -p 8 -G stringtie_merged.gtf -o
ballgown/NP300nmTiO2NP2/NP300nmTiO2NP2.gtf NP300nmTiO2NP2.bam
stringtie -e -B -p 8 -G stringtie_merged.gtf -o ballgown/NPQD1_1/NPQD1_1.gtf
NPQD1_1.bam
stringtie -e -B -p 8 -G stringtie_merged.gtf -o ballgown/NPQD1_2/NPQD1_2.gtf
NPQD1_2.bam
stringtie -e -B -p 8 -G stringtie_merged.gtf -o ballgown/NPQD2_1/NPQD2_1.gtf
NPQD2_1.bam
stringtie -e -B -p 8 -G stringtie_merged.gtf -o ballgown/NPQD2_2/NPQD2_2.gtf
NPQD2_2.bam

```

[7] Load the phenotype data for the samples.

```
Pheno_data = read.csv("phenotype_data.csv")
```

Pheno_data.csv:

id	type	state
NPHeparg1	Control	No treatment
NPHeparg2	Control	No treatment
NP3nmAuNP1	Gold	Treatment
NP3nmAuNP2	Gold	Treatment
NP15nmAuNP1	Gold	Treatment
NP15nmAuNP2	Gold	Treatment
NP80nmAuNP1	Gold	Treatment

NP80nmAuNP2	Gold	Treatment
NP12nmAgNP1	Silver	Treatment
NP12nmAgNP2	Silver	Treatment
NP40nmAgNP1	Silver	Treatment
NP40nmAgNP2	Silver	Treatment
NP80nmAgNP1	Silver	Treatment
NP80nmAgNP2	Silver	Treatment
NP25nmTiO2NP1	Titanium	Treatment
NP25nmTiO2NP2	Titanium	Treatment
NP100nmTiO2NP1	Titanium	Treatment
NP100nmTiO2NP2	Titanium	Treatment
NP300nmTiO2NP1	Titanium	Treatment
NP300nmTiO2NP2	Titanium	Treatment
NPQD1_1	QD	Treatment
NPQD1_2	QD	Treatment
NPQD2_1	QD	Treatment
NPQD2_2	QD	Treatment

[8] Read the expression data in R by Ballgown

```
bg = ballgown(dataDir="ballgown", samplePattern = "NP", pData=pheno_data)
All(pheno_data$ids == list.files ("ballgown"))
```

[9] Filter to remove low-abundance genes.

```
bg_filt = subset (bg, "rowVars(texpr(bg))>1", genomesubset=TRUE)
```

[10] Identify genes that show statistically significant differences between groups.

```
results_genes = statstest (bg_filt, feature = "gene", covariate = "state", getFC=TRUE,  
meas="FPKM")
```

[11] Sort the results from the smallest P value to the largest.

```
Results_genes = arrange (results_genes, pval)  
Write.csv (results_genes, "results_genes.csv", sep=";", "dec=", ",", row.names=FALSE)
```

[12] Showing changes in expression profiles as volcano plots.

NP3nmAuNP:

```
results_gene_3gold <- read.csv("Results_gene_3gold.csv")  
gold3Volcano <- as.data.frame(results_gene_3gold)  
gold3Volcano$log2fc <- log(gold3Volcano$fc,2)  
gold3Volcano$significant <- gold3Volcano$pval < 0.05  
gold3Volcano$Modulation <- "NS"  
gold3Volcano$Modulation <- ifelse(gold3Volcano$pval <= 0.05 &  
gold3Volcano$log2fc > 0.5, "UP",gold3Volcano$Modulation)  
gold3Volcano$Modulation <- ifelse(gold3Volcano$pval <= 0.05 &  
gold3Volcano$log2fc < -0.5, "DN",gold3Volcano$Modulation)  
gold3VolcanoPlot <- ggplot(gold3Volcano, aes(x = log2fc, y = -log10(pval), color =  
Modulation)) +  
  geom_point() +  
  geom_text(aes(label = paste0("n = ",nrow(gold3Volcano[gold3Volcano[11] ==  
"UP",])), color = "red",  
    x = 7, y = max(-log10(gold3Volcano$pval)),  
    hjust = 1, vjust = 1,  
    size = 5) +  
  geom_text(aes(label = paste0("n = ",nrow(gold3Volcano[gold3Volcano[11] ==  
"DN",])), color = "blue",  
    x = -2, y = max(-log10(gold3Volcano$pval)),  
    hjust = 1, vjust = 1,  
    size = 5) +
```

```

theme_minimal() +
labs(title = "3nm AuNP vs HepaRG Control", x = "Log2(FC)", y = "-Log(p-val)") +
scale_color_manual(values = c("blue", "gray", "red"))
(gold3VolcanoPlot)

```

NP15nmAuNP:

```

results_gene_15gold <- read.csv("Results_gene_15gold.csv")
gold15Volcano <- as.data.frame(results_gene_15gold)
gold15Volcano$log2fc <- log(gold15Volcano$fc, 2)
gold15Volcano$significant <- gold15Volcano$pval < 0.05
gold15Volcano$Modulation <- "NS"
gold15Volcano$Modulation <- ifelse(gold15Volcano$pval <= 0.05 &
gold15Volcano$log2fc > 0.5, "UP", gold15Volcano$Modulation)
gold15Volcano$Modulation <- ifelse(gold15Volcano$pval <= 0.05 &
gold15Volcano$log2fc < -0.5, "DN", gold15Volcano$Modulation)
gold15VolcanoPlot <- ggplot(gold15Volcano, aes(x = log2fc, y = -log10(pval), color =
Modulation)) +
  geom_point() +
  geom_text(aes(label = paste0("n = ", nrow(gold15Volcano[gold15Volcano[11] ==
"UP", ]))), color = "red",
    x = 5, y = max(-log10(gold15Volcano$pval)),
    hjust = 1, vjust = 1,
    size = 5) +
  geom_text(aes(label = paste0("n = ", nrow(gold15Volcano[gold15Volcano[11] ==
"DN", ]))), color = "blue",
    x = -1, y = max(-log10(gold15Volcano$pval)),
    hjust = 1, vjust = 1,
    size = 5) +
  theme_minimal() +
  labs(title = "15nm AuNP vs HepaRG Control", x = "Log2(FC)", y = "-Log(p-val)") +
  scale_color_manual(values = c("blue", "grey", "red"))

```

(gold15VolcanoPlot)

NP80nmAuNP:

```
results_gene_80gold <- read.csv("Results_gene_80gold.csv")
gold80Volcano <- as.data.frame(results_gene_80gold)
gold80Volcano$log2fc <- log(gold80Volcano$fc,2)
gold80Volcano$significant <- gold80Volcano$pval < 0.05
gold80Volcano$Modulation <- "NS"
gold80Volcano$Modulation <- ifelse(gold80Volcano$pval <= 0.05 &
gold80Volcano$log2fc > 0.5, "UP",gold80Volcano$Modulation)
gold80Volcano$Modulation <- ifelse(gold80Volcano$pval <= 0.05 &
gold80Volcano$log2fc < -0.5, "DN",gold80Volcano$Modulation)
gold80VolcanoPlot <- ggplot(gold80Volcano, aes(x = log2fc, y = -log10(pval), color =
Modulation)) +
  geom_point() +
  geom_text(aes(label = paste0("n = ",nrow(gold80Volcano[gold80Volcano[11] ==
"UP",])), color = "red",
    x = 3, y = max(-log10(gold80Volcano$pval)),
    hjust = 1, vjust = 1,
    size = 5) +
  geom_text(aes(label = paste0("n = ",nrow(gold80Volcano[gold80Volcano[11] ==
"DN",])), color = "blue",
    x = -2, y = max(-log10(gold80Volcano$pval)),
    hjust = 1, vjust = 1,
    size = 5) +
  theme_minimal() +
  labs(title = "80nm AuNP vs HepaRG Control", x = "Log2(FC)", y = "-Log(p-val)") +
  scale_color_manual(values = c("blue", "grey", "red"))
(gold80VolcanoPlot)
```

NP12nmAgNP:

```
results_gene_Ag12 <- read.csv("Results_gene_Ag12nm.csv")
```

```

Ag12Volcano <- as.data.frame(results_gene_Ag12)
Ag12Volcano$log2fc<- log(Ag12Volcano$fc,2)
Ag12Volcano$significant <- Ag12Volcano$pval < 0.05
Ag12Volcano$Modulation <- "NS"
Ag12Volcano$Modulation <- ifelse(Ag12Volcano$pval <= 0.05 &
Ag12Volcano$log2fc > 0.5, "UP",Ag12Volcano$Modulation)
Ag12Volcano$Modulation <- ifelse(Ag12Volcano$pval <= 0.05 &
Ag12Volcano$log2fc < -0.5, "DN",Ag12Volcano$Modulation)
Ag12VolcanoPlot <- ggplot(Ag12Volcano, aes(x = log2fc, y = -log10(pval), color =
Modulation)) +
  geom_point() +
  geom_text(aes(label = paste0("n = ",nrow(Ag12Volcano[Ag12Volcano[11] ==
"UP",])), color = "red",
    x = 5, y = max(-log10(Ag12Volcano$pval)),
    hjust = 1, vjust = 1,
    size = 5) +
  geom_text(aes(label = paste0("n = ",nrow(Ag12Volcano[Ag12Volcano[11] ==
"DN",])), color = "blue",
    x = -2, y = max(-log10(Ag12Volcano$pval)),
    hjust = 1, vjust = 1,
    size = 5) +
  theme_minimal() +
  labs(title = "12nm AgNP vs HepaRG Control", x = "Log2(FC)", y = "-Log(p-val)") +
  scale_color_manual(values = c("blue", "gray", "red"))
(Ag12VolcanoPlot)

```

NP40nmAgNP:

```

results_gene_Ag40 <- read.csv("Results_gene_Ag40nm.csv")
Ag40Volcano <- as.data.frame(results_gene_Ag40)
Ag40Volcano$log2fc<- log(Ag40Volcano$fc,2)
Ag40Volcano$significant <- Ag40Volcano$pval < 0.05
Ag40Volcano$Modulation <- "NS"

```

```

Ag40Volcano$Modulation <- ifelse(Ag40Volcano$pval <= 0.05 &
Ag40Volcano$log2fc > 0.5, "UP",Ag40Volcano$Modulation)
Ag40Volcano$Modulation <- ifelse(Ag40Volcano$pval <= 0.05 &
Ag40Volcano$log2fc < -0.5, "DN",Ag40Volcano$Modulation)
Ag40VolcanoPlot <- ggplot(Ag40Volcano, aes(x = log2fc, y = -log10(pval), color =
Modulation)) +
  geom_point() +
  geom_text(aes(label = paste0("n = ",nrow(Ag40Volcano[Ag40Volcano[11] ==
"UP",])), color = "red",
    x = 7, y = max(-log10(Ag40Volcano$pval)),
    hjust = 1, vjust = 1,
    size = 5) +
  geom_text(aes(label = paste0("n = ",nrow(Ag40Volcano[Ag40Volcano[11] ==
"DN",])), color = "blue",
    x = -3, y = max(-log10(Ag40Volcano$pval)),
    hjust = 1, vjust = 1,
    size = 5) +
  theme_minimal() +
  labs(title = "40nm AgNP vs HepaRG Control", x = "Log2(FC)", y = "-Log(p-val)") +
  scale_color_manual(values = c("blue", "gray", "red"))
(Ag40VolcanoPlot)

```

NP80nmAgNP:

```

results_gene_Ag80 <- read.csv("Results_gene_Ag80nm.csv")
Ag80Volcano <- as.data.frame(results_gene_Ag80)
Ag80Volcano$log2fc<- log(Ag80Volcano$fc,2)
Ag80Volcano$significant <- Ag80Volcano$pval < 0.05
Ag80Volcano$Modulation <- "NS"
Ag80Volcano$Modulation <- ifelse(Ag80Volcano$pval <= 0.05 &
Ag80Volcano$log2fc > 0.5, "UP",Ag80Volcano$Modulation)
Ag80Volcano$Modulation <- ifelse(Ag80Volcano$pval <= 0.05 &
Ag80Volcano$log2fc < -0.5, "DN",Ag80Volcano$Modulation)

```

```

Ag80VolcanoPlot <- ggplot(Ag80Volcano, aes(x = log2fc, y = -log10(pval), color =
Modulation)) +
  geom_point() +
  geom_text(aes(label = paste0("n = ",nrow(Ag80Volcano[Ag80Volcano[11] ==
"UP",])), color = "red",
    x = 7, y = max(-log10(Ag80Volcano$pval)),
    hjust = 1, vjust = 1,
    size = 5) +
  geom_text(aes(label = paste0("n = ",nrow(Ag80Volcano[Ag80Volcano[11] ==
"DN",])), color = "blue",
    x = -3, y = max(-log10(Ag80Volcano$pval)),
    hjust = 1, vjust = 1,
    size = 5) +
  theme_minimal() +
  labs(title = "80nm AgNP vs HepaRG Control", x = "Log2(FC)", y = "-Log(p-val)") +
  scale_color_manual(values = c("blue", "gray", "red"))
(Ag80VolcanoPlot)

```

NP25nmTiO₂NP:

```

results_gene_Ti25 <- read.csv("Results_gene_Ti25nm.csv")
Ti25Volcano <- as.data.frame(results_gene_Ti25)
Ti25Volcano$log2fc <- log(Ti25Volcano$fc,2)
Ti25Volcano$significant <- Ti25Volcano$pval < 0.05
Ti25Volcano$Modulation <- "NS"
Ti25Volcano$Modulation <- ifelse(Ti25Volcano$pval <= 0.05 & Ti25Volcano$log2fc >
0.5, "UP",Ti25Volcano$Modulation)
Ti25Volcano$Modulation <- ifelse(Ti25Volcano$pval <= 0.05 & Ti25Volcano$log2fc <
-0.5, "DN",Ti25Volcano$Modulation)
Ti25VolcanoPlot <- ggplot(Ti25Volcano, aes(x = log2fc, y = -log10(pval), color =
Modulation)) +
  geom_point() +

```

```

geom_text(aes(label = paste0("n = ",nrow(Ti25Volcano[Ti25Volcano[11] ==
"UP",])), color = "red",
          x = 5, y = max(-log10(Ti25Volcano$pval)),
          hjust = 1, vjust = 1,
          size = 5) +
geom_text(aes(label = paste0("n = ",nrow(Ti25Volcano[Ti25Volcano[11] ==
"DN",])), color = "blue",
          x = -3, y = max(-log10(Ti25Volcano$pval)),
          hjust = 1, vjust = 1,
          size = 5) +
theme_minimal() +
labs(title = "25nm TiO2NP vs HepaRG Control", x = "Log2(FC)", y = "-Log(p-val)") +
scale_color_manual(values = c("blue", "gray", "red"))
(Ti25VolcanoPlot)

```

NP100nmTiO2NP:

```

results_gene_Ti100 <- read.csv("Results_gene_Ti100nm.csv")
Ti100Volcano <- as.data.frame(results_gene_Ti100)
Ti100Volcano$log2fc <- log(Ti100Volcano$fc,2)
Ti100Volcano$significant <- Ti100Volcano$pval < 0.05
Ti100Volcano$Modulation <- "NS"
Ti100Volcano$Modulation <- ifelse(Ti100Volcano$pval <= 0.05 &
Ti100Volcano$log2fc > 0.5, "UP",Ti100Volcano$Modulation)
Ti100Volcano$Modulation <- ifelse(Ti100Volcano$pval <= 0.05 &
Ti100Volcano$log2fc < -0.5, "DN",Ti100Volcano$Modulation)
Ti100VolcanoPlot <- ggplot(Ti100Volcano, aes(x = log2fc, y = -log10(pval), color =
Modulation)) +
  geom_point() +
  geom_text(aes(label = paste0("n = ",nrow(Ti100Volcano[Ti100Volcano[11] ==
"UP",])), color = "red",
          x = 5, y = max(-log10(Ti100Volcano$pval)),

```

```

      hjust = 1, vjust = 1,
      size = 5) +
  geom_text(aes(label = paste0("n = ",nrow(Ti100Volcano[Ti100Volcano[11] ==
"DN",])), color = "blue",
      x = -5, y = max(-log10(Ti100Volcano$pval)),
      hjust = 1, vjust = 1,
      size = 5) +
  theme_minimal() +
  labs(title = "100nm TiO2NP vs HepaRG Control", x = "Log2(FC)", y = "-Log(p-
val)")+
  scale_color_manual(values = c("blue", "gray", "red"))
(Ti100VolcanoPlot)

```

NP300nmTiO2NP:

```

results_gene_Ti300 <- read.csv("Results_gene_Ti300nm.csv")
Ti300Volcano <- as.data.frame(results_gene_Ti300)
Ti300Volcano$log2fc<- log(Ti300Volcano$fc,2)
Ti300Volcano$significant <- Ti300Volcano$pval < 0.05
Ti300Volcano$Modulation <- "NS"
Ti300Volcano$Modulation <- ifelse(Ti300Volcano$pval <= 0.05 &
Ti300Volcano$log2fc > 0.5, "UP",Ti300Volcano$Modulation)
Ti300Volcano$Modulation <- ifelse(Ti300Volcano$pval <= 0.05 &
Ti300Volcano$log2fc < -0.5, "DN",Ti300Volcano$Modulation)
Ti300VolcanoPlot <- ggplot(Ti300Volcano, aes(x = log2fc, y = -log10(pval), color =
Modulation)) +
  geom_point() +
  geom_text(aes(label = paste0("n = ",nrow(Ti300Volcano[Ti300Volcano[11] ==
"UP",])), color = "red",
      x = 6, y = max(-log10(Ti300Volcano$pval)),
      hjust = 1, vjust = 1,
      size = 5) +

```

```

geom_text(aes(label = paste0("n = ",nrow(Ti300Volcano[Ti300Volcano[11] ==
"DN",])), color = "blue",
          x = -3, y = max(-log10(Ti300Volcano$pval)),
          hjust = 1, vjust = 1,
          size = 5) +
theme_minimal() +
labs(title = "300nm TiO2NP vs HepaRG Control", x = "Log2(FC)", y = "-Log(p-
val)")+
scale_color_manual(values = c("blue", "gray", "red"))
(Ti300VolcanoPlot)

```

NPQD1:

```

results_gene_QD1 <- read.csv("Results_gene_QD1.csv")
QD1Volcano <- as.data.frame(results_gene_QD1)
QD1Volcano$log2fc<- log(QD1Volcano$fc,2)
QD1Volcano$significant <- QD1Volcano$pval < 0.05
QD1Volcano$Modulation <- "NS"
QD1Volcano$Modulation <- ifelse(QD1Volcano$pval <= 0.05 & QD1Volcano$log2fc
> 0.5, "UP",QD1Volcano$Modulation)
QD1Volcano$Modulation <- ifelse(QD1Volcano$pval <= 0.05 & QD1Volcano$log2fc
< -0.5, "DN",QD1Volcano$Modulation)

QD1VolcanoPlot <- ggplot(QD1Volcano, aes(x = log2fc, y = -log10(pval), color =
Modulation)) +

```

```

geom_point() +
geom_text(aes(label = paste0("n = ",nrow(QD1Volcano[QD1Volcano[11] ==
"UP",])), color = "red",
          x = 6, y = max(-log10(QD1Volcano$pval)),
          hjust = 1, vjust = 1,
          size = 5) +
geom_text(aes(label = paste0("n = ",nrow(QD1Volcano[QD1Volcano[11] ==
"DN",])), color = "blue",

```

```

x = -3, y = max(-log10(QD1Volcano$pval)),
hjust = 1, vjust = 1,
size = 5) +
theme_minimal() +
labs(title = "QD1 vs HepaRG Control", x = "Log2(FC)", y = "-Log(p-val)") +
scale_color_manual(values = c("blue", "gray", "red"))
(QD1VolcanoPlot)

```

NPQD2:

```

results_gene_QD2 <- read.csv("Results_gene_QD2.csv")
QD2Volcano <- as.data.frame(results_gene_QD2)
QD2Volcano$log2fc <- log(QD2Volcano$fc, 2)
QD2Volcano$significant <- QD2Volcano$pval < 0.05
QD2Volcano$Modulation <- "NS"
QD2Volcano$Modulation <- ifelse(QD2Volcano$pval <= 0.05 & QD2Volcano$log2fc
> 0.5, "UP", QD2Volcano$Modulation)
QD2Volcano$Modulation <- ifelse(QD2Volcano$pval <= 0.05 & QD2Volcano$log2fc
< -0.5, "DN", QD2Volcano$Modulation)
QD2VolcanoPlot <- ggplot(QD2Volcano, aes(x = log2fc, y = -log10(pval), color =
Modulation)) +
  geom_point() +
  geom_text(aes(label = paste0("n = ", nrow(QD2Volcano[QD2Volcano[11] ==
"UP", ]))), color = "red",
x = 7, y = max(-log10(QD2Volcano$pval)),
hjust = 1, vjust = 1,
size = 5) +
  geom_text(aes(label = paste0("n = ", nrow(QD2Volcano[QD2Volcano[11] ==
"DN", ]))), color = "blue",
x = -4, y = max(-log10(QD2Volcano$pval)),
hjust = 1, vjust = 1,
size = 5) +
  theme_minimal() +

```

```
labs(title = "QD2 vs HepaRG Control", x = "Log2(FC)", y = "-Log(p-val)") +
scale_color_manual(values = c("blue", "gray", "red"))
(QD2VolcanoPlot)
```

Volcano Plot Grid:

```
grid.arrange(gold3VolcanoPlot, gold15VolcanoPlot, gold80VolcanoPlot,
Ag12VolcanoPlot, Ag40VolcanoPlot, Ag80VolcanoPlot, Ti25VolcanoPlot,
Ti100VolcanoPlot, Ti300VolcanoPlot, QD1VolcanoPlot, QD2VolcanoPlot, ncol = 3,
nrow=4)
```

[13] Plotting Venn diagrams for the DEGs.

Venn for gold nanoparticles:

```
sig_Au3nm <- read.csv("Results_gene_3gold.csv")
sig_Au3nm <- sig_Au3nm[sig_Au3nm$pval < 0.05, ]
sig_Au3nm <- sig_Au3nm[sig_Au3nm$logFC > 0.5 | sig_Au3nm$logFC < -0.5,]
sig_Au3nm <- as.data.frame(sig_Au3nm)
```

```
sig_Au15nm <- read.csv("Results_gene_15gold.csv")
sig_Au15nm <- sig_Au15nm[sig_Au15nm$pval < 0.05, ]
sig_Au15nm <- sig_Au15nm[sig_Au15nm$logFC > 0.5 | sig_Au15nm$logFC < -0.5,]
sig_Au15nm <- as.data.frame(sig_Au15nm)
```

```
sig_Au80nm <- read.csv("Results_gene_80gold.csv")
sig_Au80nm <- sig_Au80nm[sig_Au80nm$pval < 0.05, ]
sig_Au80nm <- sig_Au80nm[sig_Au80nm$logFC > 0.5 | sig_Au80nm$logFC < -0.5,]
sig_Au80nm <- as.data.frame(sig_Au80nm)
```

```
MyVennDiagram <- venn.diagram(
x = list(rownames(sig_Au3nm), rownames(sig_Au15nm), rownames(sig_Au80nm)),
category.names = c("3 nm", "15nm", "80 nm"),
```

```

main = "Gold vs HepaRG Control",
main.fontfamily = "sans",
main.cex = 2,
filename = NULL,
fill = c("#ffcab0", "#99d1b5", "red"),
lwd = 1,
lty = 'blank',
cex = 1.5,
fontfamily = "sans",
cat.cex = 1.5,
cat.default.pos = "outer",
cat.pos = c(-40, 40, 80),
cat.dist = c(0.05, 0.05, 0.05),
cat.fontfamily = "sans")

grid.newpage()
grid.draw(MyVennDiagram)

```

Venn for silver nanoparticles:

```

sig_Ag12nm <- read.csv("Results_gene_Ag12nm.csv")
sig_Ag12nm <- sig_Ag12nm[sig_Ag12nm$pval < 0.05, ]
sig_Ag12nm <- sig_Ag12nm[sig_Ag12nm$logFC > 0.5 | sig_Ag12nm$logFC < -0.5,]
sig_Ag12nm <- as.data.frame(sig_Ag12nm)

```

```

sig_Ag40nm <- read.csv("Results_gene_Ag40nm.csv")
sig_Ag40nm <- sig_Ag40nm[sig_Ag40nm$pval < 0.05, ]
sig_Ag40nm <- sig_Ag40nm[sig_Ag40nm$logFC > 0.5 | sig_Ag40nm$logFC < -0.5,]
sig_Ag40nm <- as.data.frame(sig_Ag40nm)

```

```

sig_Ag80nm <- read.csv("Results_gene_Ag80nm.csv")
sig_Ag80nm <- sig_Ag80nm[sig_Ag80nm$pval < 0.05, ]

```

```
sig_Ag80nm <- sig_Ag80nm[sig_Ag80nm$logFC > 0.5 | sig_Ag80nm$logFC < -0.5,]
sig_Ag80nm <- as.data.frame(sig_Ag80nm)
```

```
MyVennDiagram <- venn.diagram(
  x = list(rownames(sig_Ag12nm), rownames(sig_Ag40nm), rownames(sig_Ag80nm)),
  category.names = c("12 nm", "40nm", "80 nm"),
  main = "Silver vs HepaRG Control",
  main.fontfamily = "sans",
  main.cex = 2,
  filename = NULL,
  fill = c("#ffcab0", "#99d1b5", "red"),
  lwd = 1,
  lty = 'blank',
  cex = 1.5,
  fontfamily = "sans",
  cat.cex = 1.5,
  cat.default.pos = "outer",
  cat.pos = c(-40, 40, 100),
  cat.dist = c(0.05, 0.05, 0.05),
  cat.fontfamily = "sans",
  disable.logging = TRUE)
grid.newpage()
grid.draw(MyVennDiagram)
```

Venn for titanium nanoparticles:

```
sig_Ti25nm <- read.csv("Results_gene_Ti25nm.csv")
sig_Ti25nm <- sig_Ti25nm[sig_Ti25nm$pval < 0.05, ]
sig_Ti25nm <- sig_Ti25nm[sig_Ti25nm$logFC > 0.5 | sig_Ti25nm$logFC < -0.5,]
sig_Ti25nm <- as.data.frame(sig_Ti25nm)
```

```
sig_Ti100nm <- read.csv("Results_gene_Ti100nm.csv")
```

```

sig_Ti100nm <- sig_Ti100nm[sig_Ti100nm$pval < 0.05, ]
sig_Ti100nm <- sig_Ti100nm[sig_Ti100nm$logFC > 0.5 | sig_Ti100nm$logFC < -0.5,]
sig_Ti100nm <- as.data.frame(sig_Ti100nm)

sig_Ti300nm <- read.csv("Results_gene_Ti300nm.csv")
sig_Ti300nm <- sig_Ti300nm[sig_Ti300nm$pval < 0.05, ]
sig_Ti300nm <- sig_Ti300nm[sig_Ti300nm$logFC > 0.5 | sig_Ti300nm$logFC < -0.5,]
sig_Ti300nm <- as.data.frame(sig_Ti300nm)

MyVennDiagram <- venn.diagram(
  x = list(rownames(sig_Ti25nm), rownames(sig_Ti100nm), rownames(sig_Ti300nm)),
  category.names = c("25 nm", "100nm", "300 nm"),
  main = "Titanium vs HepaRG Control",
  main.fontfamily = "sans",
  main.cex = 2,
  filename = NULL,
  fill = c("#ffcab0", "#99d1b5", "red"),
  lwd = 1.5,
  lty = 'blank',
  cex = 1.5,
  fontfamily = "sans",
  cat.cex = 1.5,
  cat.default.pos = "outer",
  cat.pos = c(-40, 40, 100),
  cat.dist = c(0.05, 0.05, 0.05),
  cat.fontfamily = "sans",
  disable.logging = TRUE)
grid.newpage()
grid.draw(MyVennDiagram)

```

Venn for quantum dots:

```

sig_QD1 <- read.csv("Results_gene_QD1.csv")
sig_QD1 <- sig_QD1[sig_QD1$pval < 0.05, ]
sig_QD1 <- sig_QD1[sig_QD1$logFC > 0.5 | sig_QD1$logFC < -0.5,]
sig_QD1 <- as.data.frame(sig_QD1)
sig_QD2 <- read.csv("Results_gene_QD2.csv")
sig_QD2 <- sig_QD2[sig_QD2$pval < 0.05, ]
sig_QD2 <- sig_QD2[sig_QD2$logFC > 0.5 | sig_QD2$logFC < -0.5,]
sig_QD2 <- as.data.frame(sig_QD2)

```

```

MyVennDiagram <- venn.diagram(
  x = list(rownames(sig_QD1), rownames(sig_QD2)),
  category.names = c("QD1", "QD2"),
  main = "Quantum Dots vs HepaRG Control",
  main.fontfamily = "sans",
  main.cex = 2,
  filename = NULL,
  fill = c("red", "#99d1b5"),
  lwd = 1.5,
  lty = 'blank',
  cex = 1.5,
  fontfamily = "sans",
  cat.cex = 1.5,
  cat.default.pos = "outer",
  cat.pos = c(-30, 30),
  cat.dist = c(0.05, 0.05),
  cat.fontfamily = "sans",
  disable.logging = TRUE)
grid.newpage()
grid.draw(MyVennDiagram)

```

Venn diagrams of common genes for each material:

```
gold_toplam= rbind(sig_Au3nm, sig_Au15nm, sig_Au80nm)
gold_toplam= gold_toplam[gold_toplam$geneNames != ".",]
gold_toplam= unique(gold_toplam$geneNames)
```

```
silver_toplam= rbind(sig_Ag12nm, sig_Ag40nm, sig_Ag80nm)
silver_toplam= silver_toplam[silver_toplam$geneNames != ".",]
silver_toplam= unique(silver_toplam$geneNames)
```

```
titanium_toplam= rbind(sig_Ti25nm, sig_Ti100nm, sig_Ti300nm)
titanium_toplam= titanium_toplam[titanium_toplam$geneNames != ".",]
titanium_toplam= unique(titanium_toplam$geneNames)
```

```
qd_toplam= rbind(sig_QD1, sig_QD2)
qd_toplam= qd_toplam[qd_toplam$geneNames != ".",]
qd_toplam= unique(qd_toplam$geneNames)
```

```
MyVennDiagram <- venn.diagram(
  x = list(gold_toplam, silver_toplam, titanium_toplam, qd_toplam),
  category.names = c("Gold", "Silver", "Titanium", "Quantum Dot"),
  main = "Common Genes",
  main.fontfamily = "sans",
  main.cex = 3,
  filename = NULL,
  fill = c("red", "#99d1b5", "blue", "pink"),
  lwd = 2,
  lty = 'blank',
  cex = 2,
  fontfamily = "sans",
  cat.cex = 2,
  cat.default.pos = "outer",
```

```

cat.pos = c(-30, 20, -5, 40),
cat.dist = c(0.2, 0.25, 0.15, 0.2),
cat.fontfamily = "sans",
disable.logging = TRUE)
grid.newpage()

```

[14] Construction of Heatmaps for each NP with various gene sets.

REACTOME_METAL_ION_SLC_TRANSPORTERS

```

msigdb1 <- msigdb(species = "human",category = "C2")
msigdb1 <- msigdb1[msigdb1$gs_subcat == "CP:REACTOME",]
msigdb1 <- msigdb1[msigdb1$gs_name ==
"REACTOME_METAL_ION_SLC_TRANSPORTERS", ]
res3Au <- read.csv("Results_gene_3gold.csv")
res3Au <- res3Au[res3Au$geneNames != ".",]
M1_3Au <- res3Au[res3Au$geneNames %in% msigdb1$gene_symbol,]
M1_3Au <- M1_3Au[order(M1_3Au$geneNames), ]

```

```

res15Au <- read.csv("Results_gene_15gold.csv")
res15Au <- res15Au[res15Au$geneNames != ".",]
M1_15Au <- res15Au[res15Au$geneNames %in% msigdb1$gene_symbol,]
M1_15Au <- M1_15Au[order(M1_15Au$geneNames), ]

```

```

res80Au <- read.csv("Results_gene_80gold.csv")

```

```

res80Au <- res80Au[res80Au$geneNames != "."]

M1_80Au <- res80Au[res80Au$geneNames %in% msigdb_r1$gene_symbol,]

M1_80Au <- M1_80Au[order(M1_80Au$geneNames), ]

M1_15Au <- M1_15Au[M1_15Au$geneNames %in% M1_3Au$geneNames,]

M1_15Au <- M1_15Au[M1_15Au$geneNames %in% M1_80Au$geneNames,]

M1_3Au <- M1_3Au[M1_3Au$geneNames %in% M1_15Au$geneNames,]

M1_80Au <- M1_80Au[M1_80Au$geneNames %in% M1_15Au$geneNames,]


Au <- cbind(M1_3Au$logFC,M1_15Au$logFC,M1_80Au$logFC)

Au <- as.data.frame(Au)

rownames(Au) <- M1_15Au$geneNames

colnames(Au) <- c("Au3nm","Au15nm","Au80nm")

Au_p <- cbind(M1_3Au$pval,M1_15Au$pval,M1_80Au$pval)

Au_p <- as.data.frame(Au_p)

rownames(Au_p) <- M1_15Au$geneNames

colnames(Au_p) <- c("3nm","15nm","80nm")

Au <- as.matrix(Au)

ht <- Heatmap(Au,

               cluster_columns = F,

```

```

show_row_names = T,

name = "Log2FC",

column_title = "METAL ION SLC TRANSPORTERS",

column_names_gp = grid::gpar(fontsize = 14),

column_title_gp = grid::gpar(fontsize = 14, fontface = "bold"),

colorRamp2(c(-2, 0, 2), c("#2071f5", "white", "#ff4a50")))

Au_p <- as.matrix(Au_p)

ht_p <- Heatmap(Au_p,

  name = "p value",

  show_row_names = T,

  cluster_columns = F,

  cluster_rows = F,

  width = unit(1, "cm"),

  colorRamp2(c(0, 1), c("#3d0c5c", "white")))

ht + ht_p

REACTOME_RESPONSE_TO_METAL_IONS

msigdb_r2 <- msigdb_r2[species == "human",category == "C2"]

msigdb_r2 <- msigdb_r2[msigdb_r2$gs_subcat == "CP:REACTOME",]

msigdb_r2 <- msigdb_r2[msigdb_r2$gs_name ==
"REACTOME_RESPONSE_TO_METAL_IONS", ]

```

```

res3Au <- read.csv("Results_gene_3gold.csv")

res3Au <- res3Au[res3Au$geneNames != ".",]

M1_3Au <- res3Au[res3Au$geneNames %in% msigdb_2$gene_symbol,]

M1_3Au <- M1_3Au[order(M1_3Au$geneNames), ]

res15Au <- read.csv("Results_gene_15gold.csv")

res15Au <- res15Au[res15Au$geneNames != ".",]

M1_15Au <- res15Au[res15Au$geneNames %in% msigdb_2$gene_symbol,]

M1_15Au <- M1_15Au[order(M1_15Au$geneNames), ]

res80Au <- read.csv("Results_gene_80gold.csv")

res80Au <- res80Au[res80Au$geneNames != ".",]

M1_80Au <- res80Au[res80Au$geneNames %in% msigdb_2$gene_symbol,]

M1_80Au <- M1_80Au[order(M1_80Au$geneNames), ]

M1_15Au <- M1_15Au[M1_15Au$geneNames %in% M1_3Au$geneNames,]

M1_15Au <- M1_15Au[M1_15Au$geneNames %in% M1_80Au$geneNames,]

M1_3Au <- M1_3Au[M1_3Au$geneNames %in% M1_15Au$geneNames,]

M1_80Au <- M1_80Au[M1_80Au$geneNames %in% M1_15Au$geneNames,]

Au <- cbind(M1_3Au$logFC, M1_15Au$logFC, M1_80Au$logFC)

Au <- as.data.frame(Au)

rownames(Au) <- M1_15Au$geneNames

colnames(Au) <- c("Au3nm", "Au15nm", "Au80nm")

```

```

Au_p <- cbind(M1_3Au$pval,M1_15Au$pval,M1_80Au$pval)

Au_p <- as.data.frame(Au_p)

rownames(Au_p) <- M1_15Au$geneNames

colnames(Au_p) <- c("3nm","15nm","80nm")

Au <- as.matrix(Au)

ht <- Heatmap(Au,
  cluster_columns = F,
  show_row_names = T,
  name = "Log2FC",
  column_title = "CELLULAR RESPONSE TO METAL IONS",
  column_names_gp = grid::gpar(fontsize = 14),
  column_title_gp = grid::gpar(fontsize = 14, fontface = "bold"),
  colorRamp2(c(-2, 0, 2), c("#2071f5", "white", "#ff4a50")))

Au_p <- as.matrix(Au_p)

ht_p <- Heatmap(Au_p,
  name = "p value",
  show_row_names = T,
  cluster_columns = F,
  cluster_rows = F,
  width = unit(1, "cm"),

```

```
colorRamp2(c(0, 1), c("#3d0c5c", "white")))
```

```
ht + ht_p
```

```
KEGG_METABOLISM_OF_XENOBIOTICS_BY_CYTOCHROME_P450
```

```
msigdbr_3 <- msigdbr(species = "human",category = "C2")
```

```
msigdbr_3 <- msigdbr_3[msigdbr_3$gs_subcat == "CP:KEGG",]
```

```
msigdbr_3 <- msigdbr_3[msigdbr_3$gs_name ==  
"KEGG_METABOLISM_OF_XENOBIOTICS_BY_CYTOCHROME_P450", ]
```

```
res3Au <- read.csv("Results_gene_3gold.csv")
```

```
res3Au <- res3Au[res3Au$geneNames != ".",]
```

```
M1_3Au <- res3Au[res3Au$geneNames %in% msigdbr_3$gene_symbol,]
```

```
M1_3Au <- M1_3Au[order(M1_3Au$geneNames), ]
```

```
res15Au <- read.csv("Results_gene_15gold.csv")
```

```
res15Au <- res15Au[res15Au$geneNames != ".",]
```

```
M1_15Au <- res15Au[res15Au$geneNames %in% msigdbr_3$gene_symbol,]
```

```
M1_15Au <- M1_15Au[order(M1_15Au$geneNames), ]
```

```
res80Au <- read.csv("Results_gene_80gold.csv")
```

```
res80Au <- res80Au[res80Au$geneNames != ".",]
```

```
M1_80Au <- res80Au[res80Au$geneNames %in% msigdbr_3$gene_symbol,]
```

```
M1_80Au <- M1_80Au[order(M1_80Au$geneNames), ]
```

```

M1_15Au <- M1_15Au[M1_15Au$geneNames %in% M1_3Au$geneNames,]

M1_15Au <- M1_15Au[M1_15Au$geneNames %in% M1_80Au$geneNames,]

M1_3Au <- M1_3Au[M1_3Au$geneNames %in% M1_15Au$geneNames,]

M1_80Au <- M1_80Au[M1_80Au$geneNames %in% M1_15Au$geneNames,]

Au <- cbind(M1_3Au$logFC,M1_15Au$logFC,M1_80Au$logFC)

Au <- as.data.frame(Au)

rownames(Au) <- M1_15Au$geneNames

colnames(Au) <- c("Au3nm","Au15nm","Au80nm")

Au_p <- cbind(M1_3Au$pval,M1_15Au$pval,M1_80Au$pval)

Au_p <- as.data.frame(Au_p)

rownames(Au_p) <- M1_15Au$geneNames

colnames(Au_p) <- c("3nm","15nm","80nm")

Au <- as.matrix(Au)

ht <- Heatmap(Au,

               cluster_columns = F,

               show_row_names = T,

               name = "Log2FC",

               column_title = "METABOLISM OF XENOBIOTICS BY CYTOCHROME
P450",

               column_names_gp = grid::gpar(fontsize = 14),

```

```

column_title_gp = grid::gpar(fontsize = 14, fontface = "bold"),

colorRamp2(c(-2, 0, 2), c("#2071f5", "white", "#ff4a50")))

Au_p <- as.matrix(Au_p)

ht_p <- Heatmap(Au_p,

  name = "p value",

  show_row_names = T,

  cluster_columns = F,

  cluster_rows = F,

  width = unit(1, "cm"),

  colorRamp2(c(0, 1), c("#3d0c5c", "white")))

ht + ht_p

```

[15] **Hallmark gene set enrichment analysis**

```

HALLMARK_OXIDATIVE_PHOSPHORYLATION

msigdb_r4 <- msigdb_r4[species = "human",category = "H"]

msigdb_r4 <- msigdb_r4[msigdb_r4$gs_name ==
"HALLMARK_OXIDATIVE_PHOSPHORYLATION", ]

res3Au <- read.csv("Results_gene_3gold.csv")

res3Au <- res3Au[res3Au$geneNames != ".",]

M1_3Au <- res3Au[res3Au$geneNames %in% msigdb_r4$gene_symbol,]

M1_3Au <- M1_3Au[order(M1_3Au$geneNames), ]

```

```

res15Au <- read.csv("Results_gene_15gold.csv")

res15Au <- res15Au[res15Au$geneNames != ".",]

M1_15Au <- res15Au[res15Au$geneNames %in% msigdb_r4$gene_symbol,]

M1_15Au <- M1_15Au[order(M1_15Au$geneNames), ]

res80Au <- read.csv("Results_gene_80gold.csv")

res80Au <- res80Au[res80Au$geneNames != ".",]

M1_80Au <- res80Au[res80Au$geneNames %in% msigdb_r4$gene_symbol,]

M1_80Au <- M1_80Au[order(M1_80Au$geneNames), ]

M1_15Au <- M1_15Au[M1_15Au$geneNames %in% M1_3Au$geneNames,]

M1_15Au <- M1_15Au[M1_15Au$geneNames %in% M1_80Au$geneNames,]

M1_3Au <- M1_3Au[M1_3Au$geneNames %in% M1_15Au$geneNames,]

M1_80Au <- M1_80Au[M1_80Au$geneNames %in% M1_15Au$geneNames,]

Au <- cbind(M1_3Au$logFC,M1_15Au$logFC,M1_80Au$logFC)

Au <- as.data.frame(Au)

rownames(Au) <- M1_15Au$geneNames

colnames(Au) <- c("Au3nm","Au15nm","Au80nm")

Au_p <- cbind(M1_3Au$pval,M1_15Au$pval,M1_80Au$pval)

Au_p <- as.data.frame(Au_p)

rownames(Au_p) <- M1_15Au$geneNames

colnames(Au_p) <- c("3nm","15nm","80nm")

```

```

Au <- as.matrix(Au)

ht <- Heatmap(Au,

  cluster_columns = F,

  show_row_names = T,

  name = "Log2FC",

  column_title = "HALLMARK OXIDATIVE PHOSPHORYLATION",

  column_names_gp = grid::gpar(fontsize = 14),

  column_title_gp = grid::gpar(fontsize = 14, fontface = "bold"),

  colorRamp2(c(-2, 0, 2), c("#2071f5", "white", "#ff4a50")))

Au_p <- as.matrix(Au_p)

ht_p <- Heatmap(Au_p,

  name = "p value",

  show_row_names = T,

  cluster_columns = F,

  cluster_rows = F,

  width = unit(1, "cm"),

  colorRamp2(c(0, 1), c("#3d0c5c", "white")))

ht + ht_p

HALLMARK_APOPTOSIS

msigdb_r_5 <- msigdb_r(species = "human", category = "H")

```

```

msigdb_5 <- msigdb_5[msigdb_5$gs_name == "HALLMARK_APOPTOSIS", ]

res3Au <- read.csv("Results_gene_3gold.csv")

res3Au <- res3Au[res3Au$geneNames != ".",]

M1_3Au <- res3Au[res3Au$geneNames %in% msigdb_5$gene_symbol,]

M1_3Au <- M1_3Au[order(M1_3Au$geneNames), ]

res15Au <- read.csv("Results_gene_15gold.csv")

res15Au <- res15Au[res15Au$geneNames != ".",]

M1_15Au <- res15Au[res15Au$geneNames %in% msigdb_5$gene_symbol,]

M1_15Au <- M1_15Au[order(M1_15Au$geneNames), ]

res80Au <- read.csv("Results_gene_80gold.csv")

res80Au <- res80Au[res80Au$geneNames != ".",]

M1_80Au <- res80Au[res80Au$geneNames %in% msigdb_5$gene_symbol,]

M1_80Au <- M1_80Au[order(M1_80Au$geneNames), ]

M1_15Au <- M1_15Au[M1_15Au$geneNames %in% M1_3Au$geneNames,]

M1_15Au <- M1_15Au[M1_15Au$geneNames %in% M1_80Au$geneNames,]

M1_3Au <- M1_3Au[M1_3Au$geneNames %in% M1_15Au$geneNames,]

M1_80Au <- M1_80Au[M1_80Au$geneNames %in% M1_15Au$geneNames,]

Au <- cbind(M1_3Au$logFC, M1_15Au$logFC, M1_80Au$logFC)

Au <- as.data.frame(Au)

rownames(Au) <- M1_15Au$geneNames

```

```

colnames(Au) <- c("Au3nm","Au15nm","Au80nm")

Au_p <- cbind(M1_3Au$pval,M1_15Au$pval,M1_80Au$pval)

Au_p <- as.data.frame(Au_p)

rownames(Au_p) <- M1_15Au$geneNames

colnames(Au_p) <- c("3nm","15nm","80nm")

Au <- as.matrix(Au)

ht <- Heatmap(Au,

               cluster_columns = F,

               show_row_names = T,

               name = "Log2FC",

               column_title = "HALLMARK APOPTOSIS",

               column_names_gp = grid::gpar(fontsize = 14),

               column_title_gp = grid::gpar(fontsize = 14, fontface = "bold"),

               colorRamp2(c(-2, 0, 2), c("#2071f5", "white", "#ff4a50")))

Au_p <- as.matrix(Au_p)

ht_p <- Heatmap(Au_p,

                name = "p value",

                show_row_names = T,

                cluster_columns = F,

                cluster_rows = F,

```

```

width = unit(1, "cm"),

colorRamp2(c(0, 1), c("#3d0c5c", "white")))

ht + ht_p

ALONSO_METASTASIS_EMT

msigdbr_6 <- msigdbr(species = "human",category = "C2")

msigdbr_6 <- msigdbr_6[msigdbr_6$gs_subcat == "CGP",]

msigdbr_6 <- msigdbr_6[msigdbr_6$gs_name ==
"ALONSO_METASTASIS_EMT_UP", ]

msigdbr_6_ <- msigdbr(species = "human",category = "C2")

msigdbr_6_ <- msigdbr_6_[msigdbr_6_$gs_subcat == "CGP",]

msigdbr_6_ <- msigdbr_6_[msigdbr_6_$gs_name ==
"ALONSO_METASTASIS_EMT_DN", ]

msigdbr_6 <- rbind(msigdbr_6,msigdbr_6_)

res3Au <- read.csv("Results_gene_3gold.csv")

res3Au <- res3Au[res3Au$geneNames != ".",]

M1_3Au <- res3Au[res3Au$geneNames %in% msigdbr_6$gene_symbol,]

M1_3Au <- M1_3Au[order(M1_3Au$geneNames), ]

res15Au <- read.csv("Results_gene_15gold.csv")

res15Au <- res15Au[res15Au$geneNames != ".",]

M1_15Au <- res15Au[res15Au$geneNames %in% msigdbr_6$gene_symbol,]

M1_15Au <- M1_15Au[order(M1_15Au$geneNames), ]

```

```

res80Au <- read.csv("Results_gene_80gold.csv")

res80Au <- res80Au[res80Au$geneNames != ".",]

M1_80Au <- res80Au[res80Au$geneNames %in% msigdb_r6$gene_symbol,]

M1_80Au <- M1_80Au[order(M1_80Au$geneNames), ]

M1_15Au <- M1_15Au[M1_15Au$geneNames %in% M1_3Au$geneNames,]

M1_15Au <- M1_15Au[M1_15Au$geneNames %in% M1_80Au$geneNames,]

M1_3Au <- M1_3Au[M1_3Au$geneNames %in% M1_15Au$geneNames,]

M1_80Au <- M1_80Au[M1_80Au$geneNames %in% M1_15Au$geneNames,]

Au <- cbind(M1_3Au$logFC,M1_15Au$logFC,M1_80Au$logFC)

Au <- as.data.frame(Au)

rownames(Au) <- M1_15Au$geneNames

colnames(Au) <- c("Au3nm","Au15nm","Au80nm")

Au_p <- cbind(M1_3Au$pval,M1_15Au$pval,M1_80Au$pval)

Au_p <- as.data.frame(Au_p)

rownames(Au_p) <- M1_15Au$geneNames

colnames(Au_p) <- c("3nm","15nm","80nm")

Au <- as.matrix(Au)

ht <- Heatmap(Au,

               cluster_columns = F,

               show_row_names = T,

```

```

name = "Log2FC",

column_title = "METASTASIS EMT",

column_names_gp = grid::gpar(fontsize = 14),

column_title_gp = grid::gpar(fontsize = 14, fontface = "bold"),

colorRamp2(c(-2, 0, 2), c("#2071f5", "white", "#ff4a50")))

Au_p <- as.matrix(Au_p)

ht_p <- Heatmap(Au_p,

  name = "p value",

  show_row_names = T,

  cluster_columns = F,

  cluster_rows = F,

  width = unit(1, "cm"),

  colorRamp2(c(0, 1), c("#3d0c5c", "white")))

```

```
ht + ht_p
```

LIVER CANCER

```

msigdb_r_7_2 <- msigdb_r(species = "human",category = "C2")

msigdb_r_7_2 <- msigdb_r_7_2[msigdb_r_7_2$gs_subcat == "CGP",]

msigdb_r_7_2 <- msigdb_r_7_2[msigdb_r_7_2$gs_name == "PATIL_LIVER_CANCER",
]

msigdb_r_7_3 <- msigdb_r(species = "human",category = "C2")

```

```

msigdb_7_3 <- msigdb_7_3[msigdb_7_3$gs_subcat == "CGP",]

msigdb_7_3 <- msigdb_7_3[msigdb_7_3$gs_name == "SMITH_LIVER_CANCER",
]

msigdb_7 <- rbind(msigdb_7_2,msigdb_7_3)

msigdb_7 <- msigdb_7[!duplicated(msigdb_7$gene_symbol), ]

res3Au <- read.csv("Results_gene_3gold.csv")

res3Au <- res3Au[res3Au$geneNames != ".",]

M1_3Au <- res3Au[res3Au$geneNames %in% msigdb_7$gene_symbol,]

M1_3Au <- M1_3Au[order(M1_3Au$geneNames), ]

M1_3Au <- M1_3Au %>%

  group_by(geneNames) %>%

  summarise(logFC = mean(logFC), pval = first(pval))

res15Au <- read.csv("Results_gene_15gold.csv")

res15Au <- res15Au[res15Au$geneNames != ".",]

M1_15Au <- res15Au[res15Au$geneNames %in% msigdb_7$gene_symbol,]

M1_15Au <- M1_15Au[order(M1_15Au$geneNames), ]

M1_15Au <- M1_15Au %>%

  group_by(geneNames) %>%

  summarise(logFC = mean(logFC), pval = first(pval))

```

```

res80Au <- read.csv("Results_gene_80gold.csv")

res80Au <- res80Au[res80Au$geneNames != ".",]

M1_80Au <- res80Au[res80Au$geneNames %in% msigdb_r7$gene_symbol,]

M1_80Au <- M1_80Au[order(M1_80Au$geneNames), ]

M1_80Au <- M1_80Au %>%

  group_by(geneNames) %>%

  summarise(logFC = mean(logFC), pval = first(pval))

M1_3Au <- M1_3Au[M1_3Au$geneNames %in% M1_80Au$geneNames,]

M1_3Au <- M1_3Au[M1_3Au$geneNames %in% M1_15Au$geneNames,]

M1_15Au <- M1_15Au[M1_15Au$geneNames %in% M1_3Au$geneNames,]

M1_80Au <- M1_80Au[M1_80Au$geneNames %in% M1_3Au$geneNames,]

Au <- cbind(M1_3Au$logFC,M1_15Au$logFC,M1_80Au$logFC)

Au <- as.data.frame(Au)

rownames(Au) <- M1_15Au$geneNames

colnames(Au) <- c("Au3nm","Au15nm","Au80nm")

Au_p <- cbind(M1_3Au$pval,M1_15Au$pval,M1_80Au$pval)

Au_p <- as.data.frame(Au_p)

rownames(Au_p) <- M1_15Au$geneNames

colnames(Au_p) <- c("3nm","15nm","80nm")

Au <- as.matrix(Au)

```

```

ht <- Heatmap(Au,

               cluster_columns = F,

               show_row_names = F,

               name = "Log2FC",

               column_title = "LIVER CANCER",

               column_names_gp = grid::gpar(fontsize = 14),

               column_title_gp = grid::gpar(fontsize = 14, fontface = "bold"),

               colorRamp2(c(-2, 0, 2), c("#2071f5", "white", "#ff4a50")))

Au_p <- as.matrix(Au_p)

ht_p <- Heatmap(Au_p,

               name = "p value",

               show_row_names = F,

               cluster_columns = F,

               cluster_rows = F,

               width = unit(1, "cm"),

               colorRamp2(c(0, 1), c("#3d0c5c", "white")))

ht + ht_p

LIVER_CANCER_UP

msigdbr_8 <- msigdbr(species = "human", category = "C2")

msigdbr_8 <- msigdbr_8[msigdbr_8$gs_subcat == "CGP",]

```

```

msigdb_8 <- msigdb_8[msigdb_8$gs_name ==
"ACEVEDO_LIVER_CANCER_UP", ]

res3Au <- read.csv("Results_gene_3gold.csv")

res3Au <- res3Au[res3Au$geneNames != ".",]

M1_3Au <- res3Au[res3Au$geneNames %in% msigdb_8$gene_symbol,]

M1_3Au <- M1_3Au[order(M1_3Au$geneNames), ]

M1_3Au <- M1_3Au %>%

  group_by(geneNames) %>%

  summarise(logFC = mean(logFC), pval = first(pval))

res15Au <- read.csv("Results_gene_15gold.csv")

res15Au <- res15Au[res15Au$geneNames != ".",]

M1_15Au <- res15Au[res15Au$geneNames %in% msigdb_8$gene_symbol,]

M1_15Au <- M1_15Au[order(M1_15Au$geneNames), ]

M1_15Au <- M1_15Au %>%

  group_by(geneNames) %>%

  summarise(logFC = mean(logFC), pval = first(pval))

res80Au <- read.csv("Results_gene_80gold.csv")

res80Au <- res80Au[res80Au$geneNames != ".",]

M1_80Au <- res80Au[res80Au$geneNames %in% msigdb_8$gene_symbol,]

M1_80Au <- M1_80Au[order(M1_80Au$geneNames), ]

```

```

M1_80Au <- M1_80Au %>%

  group_by(geneNames) %>%

  summarise(logFC = mean(logFC), pval = first(pval))

M1_3Au <- M1_3Au[M1_3Au$geneNames %in% M1_80Au$geneNames,]

M1_3Au <- M1_3Au[M1_3Au$geneNames %in% M1_15Au$geneNames,]

M1_15Au <- M1_15Au[M1_15Au$geneNames %in% M1_3Au$geneNames,]

M1_80Au <- M1_80Au[M1_80Au$geneNames %in% M1_3Au$geneNames,]

Au <- cbind(M1_3Au$logFC,M1_15Au$logFC,M1_80Au$logFC)

Au <- as.data.frame(Au)

rownames(Au) <- M1_15Au$geneNames

colnames(Au) <- c("Au3nm","Au15nm","Au80nm")

Au_p <- cbind(M1_3Au$pval,M1_15Au$pval,M1_80Au$pval)

Au_p <- as.data.frame(Au_p)

rownames(Au_p) <- M1_15Au$geneNames

colnames(Au_p) <- c("3nm","15nm","80nm")

Au <- as.matrix(Au)

ht <- Heatmap(Au,

  cluster_columns = F,

  show_row_names = F,

  name = "Log2FC",

```

```

column_title = "LIVER CANCER UP",

column_names_gp = grid::gpar(fontsize = 14),

column_title_gp = grid::gpar(fontsize = 14, fontface = "bold"),

colorRamp2(c(-2, 0, 2), c("#2071f5", "white", "#ff4a50")))

Au_p <- as.matrix(Au_p)

ht_p <- Heatmap(Au_p,

  name = "p value",

  show_row_names = F,

  cluster_columns = F,

  cluster_rows = F,

  width = unit(1, "cm"),

  colorRamp2(c(0, 1), c("#3d0c5c", "white")))

ht + ht_p

LIVER_CANCER_DN

msigdbr_9 <- msigdbr(species = "human",category = "C2")

msigdbr_9 <- msigdbr_9[msigdbr_9$gs_subcat == "CGP",]

msigdbr_9 <- msigdbr_9[msigdbr_9$gs_name ==
"ACEVEDO_LIVER_CANCER_DN", ]

res3Au <- read.csv("Results_gene_3gold.csv")

res3Au <- res3Au[res3Au$geneNames != ".",]

```

```

M1_3Au <- res3Au[res3Au$geneNames %in% msigdb_r9$gene_symbol,]

M1_3Au <- M1_3Au[order(M1_3Au$geneNames), ]

M1_3Au <- M1_3Au %>%

  group_by(geneNames) %>%

  summarise(logFC = mean(logFC), pval = first(pval))

res15Au <- read.csv("Results_gene_15gold.csv")
res15Au <- res15Au[res15Au$geneNames != ".",]

M1_15Au <- res15Au[res15Au$geneNames %in% msigdb_r9$gene_symbol,]

M1_15Au <- M1_15Au[order(M1_15Au$geneNames), ]

M1_15Au <- M1_15Au %>%

  group_by(geneNames) %>%

  summarise(logFC = mean(logFC), pval = first(pval))

res80Au <- read.csv("Results_gene_80gold.csv")
res80Au <- res80Au[res80Au$geneNames != ".",]

M1_80Au <- res80Au[res80Au$geneNames %in% msigdb_r9$gene_symbol,]

M1_80Au <- M1_80Au[order(M1_80Au$geneNames), ]

M1_80Au <- M1_80Au %>%

  group_by(geneNames) %>%

  summarise(logFC = mean(logFC), pval = first(pval))

M1_3Au <- M1_3Au[M1_3Au$geneNames %in% M1_80Au$geneNames,]

```

```

M1_3Au <- M1_3Au[M1_3Au$geneNames %in% M1_15Au$geneNames,]

M1_15Au <- M1_15Au[M1_15Au$geneNames %in% M1_3Au$geneNames,]

M1_80Au <- M1_80Au[M1_80Au$geneNames %in% M1_3Au$geneNames,]

Au <- cbind(M1_3Au$logFC,M1_15Au$logFC,M1_80Au$logFC)

Au <- as.data.frame(Au)

rownames(Au) <- M1_15Au$geneNames

colnames(Au) <- c("Au3nm","Au15nm","Au80nm")

Au_p <- cbind(M1_3Au$pval,M1_15Au$pval,M1_80Au$pval)

Au_p <- as.data.frame(Au_p)

rownames(Au_p) <- M1_15Au$geneNames

colnames(Au_p) <- c("3nm","15nm","80nm")

Au <- as.matrix(Au)

ht <- Heatmap(Au,

               cluster_columns = F,

               show_row_names = F,

               name = "Log2FC",

               column_title = "LIVER CANCER DN",

               column_names_gp = grid::gpar(fontsize = 14),

               column_title_gp = grid::gpar(fontsize = 14, fontface = "bold"),

               colorRamp2(c(-2, 0, 2), c("#2071f5", "white", "#ff4a50")))

```

```

Au_p <- as.matrix(Au_p)

ht_p <- Heatmap(Au_p,

               name = "p value",

               show_row_names = F,

               cluster_columns = F,

               cluster_rows = F,

               width = unit(1, "cm"),

               colorRamp2(c(0, 1), c("#3d0c5c", "white")))

ht + ht_p

HALLMARK_REACTIVE_OXYGEN_SPECIES_PATHWAY

msigdb_r10 <- msigdb_r(species = "human",category = "H")

msigdb_r10 <- msigdb_r10[msigdb_r10$gs_name ==
"HALLMARK_REACTIVE_OXYGEN_SPECIES_PATHWAY", ]

res3Au <- read.csv("Results_gene_3gold.csv")

res3Au <- res3Au[res3Au$geneNames != ".",]

M1_3Au <- res3Au[res3Au$geneNames %in% msigdb_r10$gene_symbol,]

M1_3Au <- M1_3Au[order(M1_3Au$geneNames), ]

res15Au <- read.csv("Results_gene_15gold.csv")

res15Au <- res15Au[res15Au$geneNames != ".",]

M1_15Au <- res15Au[res15Au$geneNames %in% msigdb_r10$gene_symbol,]

```

```

M1_15Au <- M1_15Au[order(M1_15Au$geneNames), ]

res80Au <- read.csv("Results_gene_80gold.csv")

res80Au <- res80Au[res80Au$geneNames != ".",]

M1_80Au <- res80Au[res80Au$geneNames %in% msigdb_r10$gene_symbol,]

M1_80Au <- M1_80Au[order(M1_80Au$geneNames), ]


M1_15Au <- M1_15Au[M1_15Au$geneNames %in% M1_3Au$geneNames,]

M1_15Au <- M1_15Au[M1_15Au$geneNames %in% M1_80Au$geneNames,]

M1_3Au <- M1_3Au[M1_3Au$geneNames %in% M1_15Au$geneNames,]

M1_80Au <- M1_80Au[M1_80Au$geneNames %in% M1_15Au$geneNames,]

Au <- cbind(M1_3Au$logFC,M1_15Au$logFC,M1_80Au$logFC)

Au <- as.data.frame(Au)

rownames(Au) <- M1_15Au$geneNames

colnames(Au) <- c("Au3nm","Au15nm","Au80nm")

Au_p <- cbind(M1_3Au$pval,M1_15Au$pval,M1_80Au$pval)

Au_p <- as.data.frame(Au_p)

rownames(Au_p) <- M1_15Au$geneNames

colnames(Au_p) <- c("3nm","15nm","80nm")

Au <- as.matrix(Au)

ht <- Heatmap(Au,

```

```

cluster_columns = F,

show_row_names = T,

name = "Log2FC",

column_title = "HALLMARK ROS PATHWAY",

column_names_gp = grid::gpar(fontsize = 14),

column_title_gp = grid::gpar(fontsize = 14, fontface = "bold"),

colorRamp2(c(-2, 0, 2), c("#2071f5", "white", "#ff4a50")))

Au_p <- as.matrix(Au_p)

ht_p <- Heatmap(Au_p,

  name = "p value",

  show_row_names = T,

  cluster_columns = F,

  cluster_rows = F,

  width = unit(1, "cm"),

  colorRamp2(c(0, 1), c("#3d0c5c", "white")))

ht + ht_p

```

HEATMAPS FOR SILVER NANOPARTICLES

REACTOME_METAL_ION_SLC_TRANSPORTERS

```
msigdb_r1 <- msigdb_r(species = "human", category = "C2")
```

```
msigdb_r1 <- msigdb_r1[msigdb_r1$gs_subcat == "CP:REACTOME",]
```

```

msigdb1 <- msigdb1[msigdb1$gs_name ==
"REACTOME_METAL_ION_SLC_TRANSPORTERS", ]

res12Ag <- read.csv("Results_gene_Ag12nm.csv")

res12Ag <- res12Ag[res12Ag$geneNames != ".",]

M1_12Ag <- res12Ag[res12Ag$geneNames %in% msigdb1$gene_symbol,]

M1_12Ag <- M1_12Ag[order(M1_12Ag$geneNames), ]

res40Ag <- read.csv("Results_gene_Ag40nm.csv")

res40Ag <- res40Ag[res40Ag$geneNames != ".",]

M1_40Ag <- res40Ag[res40Ag$geneNames %in% msigdb1$gene_symbol,]

M1_40Ag <- M1_40Ag[order(M1_40Ag$geneNames), ]

res80Ag <- read.csv("Results_gene_Ag80nm.csv")

res80Ag <- res80Ag[res80Ag$geneNames != ".",]

M1_80Ag <- res80Ag[res80Ag$geneNames %in% msigdb1$gene_symbol,]

M1_80Ag <- M1_80Ag[order(M1_80Ag$geneNames), ]

M1_40Ag <- M1_40Ag[M1_40Ag$geneNames %in% M1_12Ag$geneNames,]

M1_40Ag <- M1_40Ag[M1_40Ag$geneNames %in% M1_80Ag$geneNames,]

M1_12Ag <- M1_12Ag[M1_12Ag$geneNames %in% M1_40Ag$geneNames,]

M1_80Ag <- M1_80Ag[M1_80Ag$geneNames %in% M1_40Ag$geneNames,]

Au <- cbind(M1_12Ag$logFC, M1_40Ag$logFC, M1_80Ag$logFC)

Au <- as.data.frame(Au)

```

```

rownames(Au) <- M1_40Ag$geneNames

colnames(Au) <- c("Ag12nm","Ag40nm","Ag80nm")

Au_p <- cbind(M1_12Ag$pval,M1_40Ag$pval,M1_80Ag$pval)

Au_p <- as.data.frame(Au_p)

rownames(Au_p) <- M1_40Ag$geneNames

colnames(Au_p) <- c("12nm","40nm","80nm")

Au <- as.matrix(Au)

ht <- Heatmap(Au,

               cluster_columns = F,

               show_row_names = T,

               name = "Log2FC",

               column_title = "METAL ION SLC TRANSPORTERS",

               column_names_gp = grid::gpar(fontsize = 14),

               column_title_gp = grid::gpar(fontsize = 14, fontface = "bold"),

               colorRamp2(c(-2, 0, 2), c("#2071f5", "white", "#ff4a50")))

Au_p <- as.matrix(Au_p)

ht_p <- Heatmap(Au_p,

                name = "p value",

                show_row_names = T,

                cluster_columns = F,

```

```

cluster_rows = F,

width = unit(1, "cm"),

colorRamp2(c(0, 1), c("#3d0c5c", "white")))

ht + ht_p

REACTOME_RESPONSE_TO_METAL_IONS

msigdb_r2 <- msigdb_r(species = "human",category = "C2")

msigdb_r2 <- msigdb_r2[msigdb_r2$gs_subcat == "CP:REACTOME",]

msigdb_r2 <- msigdb_r2[msigdb_r2$gs_name ==
"REACTOME_RESPONSE_TO_METAL_IONS", ]

res12Ag <- read.csv("Results_gene_Ag12nm.csv")

res12Ag <- res12Ag[res12Ag$geneNames != ".",]

M1_12Ag <- res12Ag[res12Ag$geneNames %in% msigdb_r2$gene_symbol,]

M1_12Ag <- M1_12Ag[order(M1_12Ag$geneNames), ]

res40Ag <- read.csv("Results_gene_Ag40nm.csv")

res40Ag <- res40Ag[res40Ag$geneNames != ".",]

M1_40Ag <- res40Ag[res40Ag$geneNames %in% msigdb_r2$gene_symbol,]

M1_40Ag <- M1_40Ag[order(M1_40Ag$geneNames), ]

res80Ag <- read.csv("Results_gene_Ag80nm.csv")

res80Ag <- res80Ag[res80Ag$geneNames != ".",]

M1_80Ag <- res80Ag[res80Ag$geneNames %in% msigdb_r2$gene_symbol,]

```

```

M1_80Ag <- M1_80Ag[order(M1_80Ag$geneNames), ]

M1_40Ag <- M1_40Ag[M1_40Ag$geneNames %in% M1_12Ag$geneNames,]

M1_40Ag <- M1_40Ag[M1_40Ag$geneNames %in% M1_80Ag$geneNames,]

M1_12Ag <- M1_12Ag[M1_12Ag$geneNames %in% M1_40Ag$geneNames,]

M1_80Ag <- M1_80Ag[M1_80Ag$geneNames %in% M1_40Ag$geneNames,]

Au <- cbind(M1_12Ag$logFC,M1_40Ag$logFC,M1_80Ag$logFC)

Au <- as.data.frame(Au)

rownames(Au) <- M1_40Ag$geneNames

colnames(Au) <- c("Ag12nm","Ag40nm","Ag80nm")

Au_p <- cbind(M1_12Ag$pval,M1_40Ag$pval,M1_80Ag$pval)

Au_p <- as.data.frame(Au_p)

rownames(Au_p) <- M1_40Ag$geneNames

colnames(Au_p) <- c("12nm","40nm","80nm")

Au <- as.matrix(Au)

ht <- Heatmap(Au,

               cluster_columns = F,

               show_row_names = T,

               name = "Log2FC",

               column_title = "CELLULAR RESPONSE TO METAL IONS",

               column_names_gp = grid::gpar(fontsize = 14),

```

```

column_title_gp = grid::gpar(fontsize = 14, fontface = "bold"),

colorRamp2(c(-2, 0, 2), c("#2071f5", "white", "#ff4a50")))

Au_p <- as.matrix(Au_p)

ht_p <- Heatmap(Au_p,

  name = "p value",

  show_row_names = T,

  cluster_columns = F,

  cluster_rows = F,

  width = unit(1, "cm"),

  colorRamp2(c(0, 1), c("#3d0c5c", "white")))

ht + ht_p

KEGG_METABOLISM_OF_XENOBIOTICS_BY_CYTOCHROME_P450

msigdb_r3 <- msigdb_r3[species = "human",category = "C2"]

msigdb_r3 <- msigdb_r3[msigdb_r3$gs_subcat == "CP:KEGG",]

msigdb_r3 <- msigdb_r3[msigdb_r3$gs_name ==
"KEGG_METABOLISM_OF_XENOBIOTICS_BY_CYTOCHROME_P450", ]

res12Ag <- read.csv("Results_gene_Ag12nm.csv")

res12Ag <- res12Ag[res12Ag$geneNames != ".",]

M1_12Ag <- res12Ag[res12Ag$geneNames %in% msigdb_r3$gene_symbol,]

M1_12Ag <- M1_12Ag[order(M1_12Ag$geneNames), ]

```

```

res40Ag <- read.csv("Results_gene_Ag40nm.csv")

res40Ag <- res40Ag[res40Ag$geneNames != ".",]

M1_40Ag <- res40Ag[res40Ag$geneNames %in% msigdb_r3$gene_symbol,]

M1_40Ag <- M1_40Ag[order(M1_40Ag$geneNames), ]

res80Ag <- read.csv("Results_gene_Ag80nm.csv")

res80Ag <- res80Ag[res80Ag$geneNames != ".",]

M1_80Ag <- res80Ag[res80Ag$geneNames %in% msigdb_r3$gene_symbol,]

M1_80Ag <- M1_80Ag[order(M1_80Ag$geneNames), ]

M1_40Ag <- M1_40Ag[M1_40Ag$geneNames %in% M1_12Ag$geneNames,]

M1_40Ag <- M1_40Ag[M1_40Ag$geneNames %in% M1_80Ag$geneNames,]

M1_12Ag <- M1_12Ag[M1_12Ag$geneNames %in% M1_40Ag$geneNames,]

M1_80Ag <- M1_80Ag[M1_80Ag$geneNames %in% M1_40Ag$geneNames,]

Au <- cbind(M1_12Ag$logFC,M1_40Ag$logFC,M1_80Ag$logFC)

Au <- as.data.frame(Au)

rownames(Au) <- M1_40Ag$geneNames

colnames(Au) <- c("Ag12nm","Ag40nm","Ag80nm")

Au_p <- cbind(M1_12Ag$pval,M1_40Ag$pval,M1_80Ag$pval)

Au_p <- as.data.frame(Au_p)

rownames(Au_p) <- M1_40Ag$geneNames

colnames(Au_p) <- c("12nm","40nm","80nm")

```

```

Au <- as.matrix(Au)

ht <- Heatmap(Au,

  cluster_columns = F,

  show_row_names = T,

  name = "Log2FC",

  column_title = "METABOLISM OF XENOBIOTICS BY CYTOCHROME
P450",

  column_names_gp = grid::gpar(fontsize = 14),

  column_title_gp = grid::gpar(fontsize = 14, fontface = "bold"),

  colorRamp2(c(-2, 0, 2), c("#2071f5", "white", "#ff4a50")))

Au_p <- as.matrix(Au_p)

ht_p <- Heatmap(Au_p,

  name = "p value",

  show_row_names = T,

  cluster_columns = F,

  cluster_rows = F,

  width = unit(1, "cm"),

  colorRamp2(c(0, 1), c("#3d0c5c", "white")))

ht + ht_p

HALLMARK_OXIDATIVE_PHOSPHORYLATION

```

```

msigdb_4 <- msigdb(species = "human",category = "H")

msigdb_4 <- msigdb_4[msigdb_4$gs_name ==
"HALLMARK_OXIDATIVE_PHOSPHORYLATION", ]

res12Ag <- read.csv("Results_gene_Ag12nm.csv")

res12Ag <- res12Ag[res12Ag$geneNames != ".",]

M1_12Ag <- res12Ag[res12Ag$geneNames %in% msigdb_4$gene_symbol,]

M1_12Ag <- M1_12Ag[order(M1_12Ag$geneNames), ]

res40Ag <- read.csv("Results_gene_Ag40nm.csv")

res40Ag <- res40Ag[res40Ag$geneNames != ".",]

M1_40Ag <- res40Ag[res40Ag$geneNames %in% msigdb_4$gene_symbol,]

M1_40Ag <- M1_40Ag[order(M1_40Ag$geneNames), ]

res80Ag <- read.csv("Results_gene_Ag80nm.csv")

res80Ag <- res80Ag[res80Ag$geneNames != ".",]

M1_80Ag <- res80Ag[res80Ag$geneNames %in% msigdb_4$gene_symbol,]

M1_80Ag <- M1_80Ag[order(M1_80Ag$geneNames), ]

M1_40Ag <- M1_40Ag[M1_40Ag$geneNames %in% M1_12Ag$geneNames,]

M1_40Ag <- M1_40Ag[M1_40Ag$geneNames %in% M1_80Ag$geneNames,]

M1_12Ag <- M1_12Ag[M1_12Ag$geneNames %in% M1_40Ag$geneNames,]

M1_80Ag <- M1_80Ag[M1_80Ag$geneNames %in% M1_40Ag$geneNames,]

Au <- cbind(M1_12Ag$logFC,M1_40Ag$logFC,M1_80Ag$logFC)

```

```

Au <- as.data.frame(Au)

rownames(Au) <- M1_40Ag$geneNames

colnames(Au) <- c("Ag12nm","Ag40nm","Ag80nm")

Au_p <- cbind(M1_12Ag$pval,M1_40Ag$pval,M1_80Ag$pval)

Au_p <- as.data.frame(Au_p)

rownames(Au_p) <- M1_40Ag$geneNames

colnames(Au_p) <- c("12nm","40nm","80nm")

Au <- as.matrix(Au)

ht <- Heatmap(Au,

               cluster_columns = F,

               show_row_names = T,

               name = "Log2FC",

               column_title = "HALLMARK OXIDATIVE PHOSPHORYLATION",

               column_names_gp = grid::gpar(fontsize = 14),

               column_title_gp = grid::gpar(fontsize = 14, fontface = "bold"),

               colorRamp2(c(-2, 0, 2), c("#2071f5", "white", "#ff4a50")))

Au_p <- as.matrix(Au_p)

ht_p <- Heatmap(Au_p,

                name = "p value",

                show_row_names = T,

```

```

cluster_columns = F,

cluster_rows = F,

width = unit(1, "cm"),

colorRamp2(c(0, 1), c("#3d0c5c", "white")))

ht + ht_p

HALLMARK_APOPTOSIS

msigdb_r_5 <- msigdb_r(species = "human",category = "H")

msigdb_r_5 <- msigdb_r[msigdb_r$gs_name == "HALLMARK_APOPTOSIS", ]

res12Ag <- read.csv("Results_gene_Ag12nm.csv")

res12Ag <- res12Ag[res12Ag$geneNames != ".",]

M1_12Ag <- res12Ag[res12Ag$geneNames %in% msigdb_r_5$gene_symbol,]

M1_12Ag <- M1_12Ag[order(M1_12Ag$geneNames), ]

res40Ag <- read.csv("Results_gene_Ag40nm.csv")

res40Ag <- res40Ag[res40Ag$geneNames != ".",]

M1_40Ag <- res40Ag[res40Ag$geneNames %in% msigdb_r_5$gene_symbol,]

M1_40Ag <- M1_40Ag[order(M1_40Ag$geneNames), ]

res80Ag <- read.csv("Results_gene_Ag80nm.csv")

res80Ag <- res80Ag[res80Ag$geneNames != ".",]

M1_80Ag <- res80Ag[res80Ag$geneNames %in% msigdb_r_5$gene_symbol,]

M1_80Ag <- M1_80Ag[order(M1_80Ag$geneNames), ]

```

```

M1_40Ag <- M1_40Ag[M1_40Ag$geneNames %in% M1_12Ag$geneNames,]

M1_40Ag <- M1_40Ag[M1_40Ag$geneNames %in% M1_80Ag$geneNames,]

M1_12Ag <- M1_12Ag[M1_12Ag$geneNames %in% M1_40Ag$geneNames,]

M1_80Ag <- M1_80Ag[M1_80Ag$geneNames %in% M1_40Ag$geneNames,]

```

```

Au <- cbind(M1_12Ag$logFC,M1_40Ag$logFC,M1_80Ag$logFC)

Au <- as.data.frame(Au)

rownames(Au) <- M1_40Ag$geneNames

colnames(Au) <- c("Ag12nm","Ag40nm","Ag80nm")

Au_p <- cbind(M1_12Ag$pval,M1_40Ag$pval,M1_80Ag$pval)

Au_p <- as.data.frame(Au_p)

rownames(Au_p) <- M1_40Ag$geneNames

colnames(Au_p) <- c("12nm","40nm","80nm")

Au <- as.matrix(Au)

ht <- Heatmap(Au,

```

```

    cluster_columns = F,

    show_row_names = T,

    name = "Log2FC",

    column_title = "HALLMARK APOPTOSIS",

    column_names_gp = grid::gpar(fontsize = 14),

```

```

column_title_gp = grid::gpar(fontsize = 14, fontface = "bold"),

colorRamp2(c(-2, 0, 2), c("#2071f5", "white", "#ff4a50")))

Au_p <- as.matrix(Au_p)

ht_p <- Heatmap(Au_p,

  name = "p value",

  show_row_names = T,

  cluster_columns = F,

  cluster_rows = F,

  width = unit(1, "cm"),

  colorRamp2(c(0, 1), c("#3d0c5c", "white")))

ht + ht_p

```

ALONSO_METASTASIS_EMT

```

msigdbr_6 <- msigdbr(species = "human",category = "C2")

msigdbr_6 <- msigdbr_6[msigdbr_6$gs_subcat == "CGP",]

msigdbr_6 <- msigdbr_6[msigdbr_6$gs_name ==
"ALONSO_METASTASIS_EMT_UP", ]

msigdbr_6_ <- msigdbr(species = "human",category = "C2")

msigdbr_6_ <- msigdbr_6_[msigdbr_6_$gs_subcat == "CGP",]

msigdbr_6_ <- msigdbr_6_[msigdbr_6_$gs_name ==
"ALONSO_METASTASIS_EMT_DN", ]

msigdbr_6 <- rbind(msigdbr_6,msigdbr_6_)

```

```

res12Ag <- read.csv("Results_gene_Ag12nm.csv")

res12Ag <- res12Ag[res12Ag$geneNames != ".",]

M1_12Ag <- res12Ag[res12Ag$geneNames %in% msigdb_r6$gene_symbol,]

M1_12Ag <- M1_12Ag[order(M1_12Ag$geneNames), ]


res40Ag <- read.csv("Results_gene_Ag40nm.csv")

res40Ag <- res40Ag[res40Ag$geneNames != ".",]

M1_40Ag <- res40Ag[res40Ag$geneNames %in% msigdb_r6$gene_symbol,]

M1_40Ag <- M1_40Ag[order(M1_40Ag$geneNames), ]

res80Ag <- read.csv("Results_gene_Ag80nm.csv")

res80Ag <- res80Ag[res80Ag$geneNames != ".",]

M1_80Ag <- res80Ag[res80Ag$geneNames %in% msigdb_r6$gene_symbol,]

M1_80Ag <- M1_80Ag[order(M1_80Ag$geneNames), ]

M1_40Ag <- M1_40Ag[M1_40Ag$geneNames %in% M1_12Ag$geneNames,]

M1_40Ag <- M1_40Ag[M1_40Ag$geneNames %in% M1_80Ag$geneNames,]

M1_12Ag <- M1_12Ag[M1_12Ag$geneNames %in% M1_40Ag$geneNames,]

M1_80Ag <- M1_80Ag[M1_80Ag$geneNames %in% M1_40Ag$geneNames,]

Au <- cbind(M1_12Ag$logFC,M1_40Ag$logFC,M1_80Ag$logFC)

Au <- as.data.frame(Au)

rownames(Au) <- M1_40Ag$geneNames

```

```

colnames(Au) <- c("Ag12nm","Ag40nm","Ag80nm")

Au_p <- cbind(M1_12Ag$pval,M1_40Ag$pval,M1_80Ag$pval)

Au_p <- as.data.frame(Au_p)

rownames(Au_p) <- M1_40Ag$geneNames

colnames(Au_p) <- c("12nm","40nm","80nm")

Au <- as.matrix(Au)

ht <- Heatmap(Au,
              cluster_columns = F,
              show_row_names = T,
              name = "Log2FC",
              column_title = "METASTASIS EMT",
              column_names_gp = grid::gpar(fontsize = 14),
              column_title_gp = grid::gpar(fontsize = 14, fontface = "bold"),
              colorRamp2(c(-2, 0, 2), c("#2071f5", "white", "#ff4a50")))

Au_p <- as.matrix(Au_p)

ht_p <- Heatmap(Au_p,
                name = "p value",
                show_row_names = T,
                cluster_columns = F,
                cluster_rows = F,

```

```

width = unit(1, "cm"),

colorRamp2(c(0, 1), c("#3d0c5c", "white")))

ht + ht_p

LIVER CANCER

msigdbr_7_2 <- msigdbr(species = "human",category = "C2")

msigdbr_7_2 <- msigdbr_7_2[msigdbr_7_2$gs_subcat == "CGP",]

msigdbr_7_2 <- msigdbr_7_2[msigdbr_7_2$gs_name == "PATIL_LIVER_CANCER",
]

msigdbr_7_3 <- msigdbr(species = "human",category = "C2")

msigdbr_7_3 <- msigdbr_7_3[msigdbr_7_3$gs_subcat == "CGP",]

msigdbr_7_3 <- msigdbr_7_3[msigdbr_7_3$gs_name == "SMITH_LIVER_CANCER",
]

msigdbr_7 <- rbind(msigdbr_7_2,msigdbr_7_3)

msigdbr_7 <- msigdbr_7[!duplicated(msigdbr_7$gene_symbol), ]

res12Ag <- read.csv("Results_gene_Ag12nm.csv")

res12Ag <- res12Ag[res12Ag$geneNames != ".",]

M1_12Ag <- res12Ag[res12Ag$geneNames %in% msigdbr_7$gene_symbol,]

M1_12Ag <- M1_12Ag[order(M1_12Ag$geneNames), ]

M1_12Ag <- M1_12Ag %>%

group_by(geneNames) %>%

summarise(logFC = mean(logFC), pval = first(pval))

```

```

res40Ag <- read.csv("Results_gene_Ag40nm.csv")

res40Ag <- res40Ag[res40Ag$geneNames != ".",]

M1_40Ag <- res40Ag[res40Ag$geneNames %in% msigdb_r7$gene_symbol,]

M1_40Ag <- M1_40Ag[order(M1_40Ag$geneNames), ]


M1_40Ag <- M1_40Ag %>%
  group_by(geneNames) %>%
  summarise(logFC = mean(logFC), pval = first(pval))

res80Ag <- read.csv("Results_gene_Ag80nm.csv")

res80Ag <- res80Ag[res80Ag$geneNames != ".",]

M1_80Ag <- res80Ag[res80Ag$geneNames %in% msigdb_r7$gene_symbol,]

M1_80Ag <- M1_80Ag[order(M1_80Ag$geneNames), ]

M1_80Ag <- M1_80Ag %>%
  group_by(geneNames) %>%
  summarise(logFC = mean(logFC), pval = first(pval))

M1_12Ag <- M1_12Ag[M1_12Ag$geneNames %in% M1_80Ag$geneNames,]

M1_12Ag <- M1_12Ag[M1_12Ag$geneNames %in% M1_40Ag$geneNames,]

M1_40Ag <- M1_40Ag[M1_40Ag$geneNames %in% M1_12Ag$geneNames,]

M1_80Ag <- M1_80Ag[M1_80Ag$geneNames %in% M1_12Ag$geneNames,]

Au <- cbind(M1_12Ag$logFC, M1_40Ag$logFC, M1_80Ag$logFC)

```

```

Au <- as.data.frame(Au)

rownames(Au) <- M1_40Ag$geneNames

colnames(Au) <- c("Ag12nm","Ag40nm","Ag80nm")


Au_p <- cbind(M1_12Ag$pval,M1_40Ag$pval,M1_80Ag$pval)

Au_p <- as.data.frame(Au_p)

rownames(Au_p) <- M1_40Ag$geneNames

colnames(Au_p) <- c("12nm","40nm","80nm")

Au <- as.matrix(Au)


ht <- Heatmap(Au,

               cluster_columns = F,

               show_row_names = F,

               name = "Log2FC",

               column_title = "LIVER CANCER",

               column_names_gp = grid::gpar(fontsize = 14),

               column_title_gp = grid::gpar(fontsize = 14, fontface = "bold"),

               colorRamp2(c(-2, 0, 2), c("#2071f5", "white", "#ff4a50")))


Au_p <- as.matrix(Au_p)

```

```

ht_p <- Heatmap(Au_p,

                name = "p value",

                show_row_names = F,

                cluster_columns = F,

                cluster_rows = F,

                width = unit(1, "cm"),

                colorRamp2(c(0, 1), c("#3d0c5c", "white")))

ht + ht_p

HALLMARK_REACTIVE_OXYGEN_SPECIES_PATHWAY

msigdb_r10 <- msigdb_r(species = "human",category = "H")

msigdb_r10 <- msigdb_r10[msigdb_r10$gs_name ==
"HALLMARK_REACTIVE_OXYGEN_SPECIES_PATHWAY", ]

res12Ag <- read.csv("Results_gene_Ag12nm.csv")

res12Ag <- res12Ag[res12Ag$geneNames != ".",]

M1_12Ag <- res12Ag[res12Ag$geneNames %in% msigdb_r10$gene_symbol,]

M1_12Ag <- M1_12Ag[order(M1_12Ag$geneNames), ]

res40Ag <- read.csv("Results_gene_Ag40nm.csv")

res40Ag <- res40Ag[res40Ag$geneNames != ".",]

M1_40Ag <- res40Ag[res40Ag$geneNames %in% msigdb_r10$gene_symbol,]

M1_40Ag <- M1_40Ag[order(M1_40Ag$geneNames), ]

```

```

res80Ag <- read.csv("Results_gene_Ag80nm.csv")

res80Ag <- res80Ag[res80Ag$geneNames != ".",]

M1_80Ag <- res80Ag[res80Ag$geneNames %in% msigdb_r10$gene_symbol,]

M1_80Ag <- M1_80Ag[order(M1_80Ag$geneNames), ]

M1_40Ag <- M1_40Ag[M1_40Ag$geneNames %in% M1_12Ag$geneNames,]

M1_40Ag <- M1_40Ag[M1_40Ag$geneNames %in% M1_80Ag$geneNames,]

M1_12Ag <- M1_12Ag[M1_12Ag$geneNames %in% M1_40Ag$geneNames,]

M1_80Ag <- M1_80Ag[M1_80Ag$geneNames %in% M1_40Ag$geneNames,]

Au <- cbind(M1_12Ag$logFC,M1_40Ag$logFC,M1_80Ag$logFC)

Au <- as.data.frame(Au)

rownames(Au) <- M1_40Ag$geneNames

colnames(Au) <- c("Ag12nm","Ag40nm","Ag80nm")

Au_p <- cbind(M1_12Ag$pval,M1_40Ag$pval,M1_80Ag$pval)

Au_p <- as.data.frame(Au_p)

rownames(Au_p) <- M1_40Ag$geneNames

colnames(Au_p) <- c("12nm","40nm","80nm")

Au <- as.matrix(Au)

ht <- Heatmap(Au,

               cluster_columns = F,

               show_row_names = T,

```

```

name = "Log2FC",

column_title = "HALLMARK ROS PATHWAY",

column_names_gp = grid::gpar(fontsize = 14),

column_title_gp = grid::gpar(fontsize = 14, fontface = "bold"),

colorRamp2(c(-2, 0, 2), c("#2071f5", "white", "#ff4a50")))

Au_p <- as.matrix(Au_p)

ht_p <- Heatmap(Au_p,

  name = "p value",

  show_row_names = T,

  cluster_columns = F,

  cluster_rows = F,

  width = unit(1, "cm"),

  colorRamp2(c(0, 1), c("#3d0c5c", "white")))

```

ht + ht_p

Significant Gene Lists

```
res12Ag <- read.csv("Results_gene_Ag12nm.csv")
```

```
res12Ag <- res12Ag[res12Ag$geneNames != ".",]
```

```
res12Ag <- res12Ag[res12Ag$pval < 0.05,]
```

```
res40Ag <- read.csv("Results_gene_Ag40nm.csv")
```

```
res40Ag <- res40Ag[res40Ag$geneNames != ".",]
```

```
res40Ag <- res40Ag[res40Ag$pval < 0.05,]
```

```
res80Ag <- read.csv("Results_gene_Ag80nm.csv")
```

```
res80Ag <- res80Ag[res80Ag$geneNames != ".",]
```

```
res80Ag <- res80Ag[res80Ag$pval < 0.05,]
```

```
res12Ag <- res12Ag[res12Ag$logFC >= 0.5 | res12Ag$logFC <= -0.5, ]
```

```
res40Ag <- res40Ag[res40Ag$logFC >= 0.5 | res40Ag$logFC <= -0.5, ]
```

```
res80Ag <- res80Ag[res80Ag$logFC >= 0.5 | res80Ag$logFC <= -0.5, ]
```

HEATMAPS FOR TITANIUM NANOPARTICLES

REACTOME_METAL_ION_SLC_TRANSPORTERS

```
msigdb_1 <- msigdb(species = "human", category = "C2")
```

```
msigdb_1 <- msigdb_1[msigdb_1$gs_subcat == "CP:REACTOME",]
```

```
msigdb_1 <- msigdb_1[msigdb_1$gs_name ==  
"REACTOME_METAL_ION_SLC_TRANSPORTERS", ]
```

```
res25Ti <- read.csv("Results_gene_Ti25nm.csv")
```

```
res25Ti <- res25Ti[res25Ti$geneNames != ".",]
```

```
M1_25Ti <- res25Ti[res25Ti$geneNames %in% msigdb_1$gene_symbol,]
```

```
M1_25Ti <- M1_25Ti[order(M1_25Ti$geneNames), ]
```

```
res100Ti <- read.csv("Results_gene_Ti100nm.csv")
```

```
res100Ti <- res100Ti[res100Ti$geneNames != ".",]
```

```

M1_100Ti <- res100Ti[res100Ti$geneNames %in% msigdb_r1$gene_symbol,]

M1_100Ti <- M1_100Ti[order(M1_100Ti$geneNames), ]

res300Ti <- read.csv("Results_gene_Ti300nm.csv")

res300Ti <- res300Ti[res300Ti$geneNames != ".",]

M1_300Ti <- res300Ti[res300Ti$geneNames %in% msigdb_r1$gene_symbol,]

M1_300Ti <- M1_300Ti[order(M1_300Ti$geneNames), ]

M1_100Ti <- M1_100Ti[M1_100Ti$geneNames %in% M1_25Ti$geneNames,]

M1_100Ti <- M1_100Ti[M1_100Ti$geneNames %in% M1_300Ti$geneNames,]

M1_25Ti <- M1_25Ti[M1_25Ti$geneNames %in% M1_100Ti$geneNames,]

M1_300Ti <- M1_300Ti[M1_300Ti$geneNames %in% M1_100Ti$geneNames,]

Au <- cbind(M1_25Ti$logFC,M1_100Ti$logFC,M1_300Ti$logFC)

Au <- as.data.frame(Au)

rownames(Au) <- M1_100Ti$geneNames

colnames(Au) <- c("Ti25nm","Ti100nm","Ti300nm")

Au_p <- cbind(M1_25Ti$pval,M1_100Ti$pval,M1_300Ti$pval)

Au_p <- as.data.frame(Au_p)

rownames(Au_p) <- M1_100Ti$geneNames

colnames(Au_p) <- c("25nm","100nm","300nm")

Au <- as.matrix(Au)

ht <- Heatmap(Au,

```

```

cluster_columns = F,

show_row_names = T,

name = "Log2FC",

column_title = "METAL ION SLC TRANSPORTERS",

column_names_gp = grid::gpar(fontsize = 14),

column_title_gp = grid::gpar(fontsize = 14, fontface = "bold"),

colorRamp2(c(-2, 0, 2), c("#2071f5", "white", "#ff4a50")))

Au_p <- as.matrix(Au_p)

ht_p <- Heatmap(Au_p,

  name = "p value",

  show_row_names = T,

  cluster_columns = F,

  cluster_rows = F,

  width = unit(1, "cm"),

  colorRamp2(c(0, 1), c("#3d0c5c", "white")))

ht + ht_p

REACTOME_RESPONSE_TO_METAL_IONS

msigdb_r2 <- msigdb_r2[msigdb_r2$gs_subcat == "CP:REACTOME",]

```

```

msigdb_r2 <- msigdb_r2[msigdb_r2$gs_name ==
"REACTOME_RESPONSE_TO_METAL_IONS", ]

res25Ti <- read.csv("Results_gene_Ti25nm.csv")

res25Ti <- res25Ti[res25Ti$geneNames != ".",]

M1_25Ti <- res25Ti[res25Ti$geneNames %in% msigdb_r2$gene_symbol,]

M1_25Ti <- M1_25Ti[order(M1_25Ti$geneNames), ]

res100Ti <- read.csv("Results_gene_Ti100nm.csv")

res100Ti <- res100Ti[res100Ti$geneNames != ".",]

M1_100Ti <- res100Ti[res100Ti$geneNames %in% msigdb_r2$gene_symbol,]

M1_100Ti <- M1_100Ti[order(M1_100Ti$geneNames), ]

res300Ti <- read.csv("Results_gene_Ti300nm.csv")

res300Ti <- res300Ti[res300Ti$geneNames != ".",]

M1_300Ti <- res300Ti[res300Ti$geneNames %in% msigdb_r2$gene_symbol,]

M1_300Ti <- M1_300Ti[order(M1_300Ti$geneNames), ]

M1_100Ti <- M1_100Ti[M1_100Ti$geneNames %in% M1_25Ti$geneNames,]

M1_100Ti <- M1_100Ti[M1_100Ti$geneNames %in% M1_300Ti$geneNames,]

M1_25Ti <- M1_25Ti[M1_25Ti$geneNames %in% M1_100Ti$geneNames,]

M1_300Ti <- M1_300Ti[M1_300Ti$geneNames %in% M1_100Ti$geneNames,]

Au <- cbind(M1_25Ti$logFC,M1_100Ti$logFC,M1_300Ti$logFC)

Au <- as.data.frame(Au)

```

```

rownames(Au) <- M1_100Ti$geneNames

colnames(Au) <- c("Ti25nm","Ti100nm","Ti300nm")

Au_p <- cbind(M1_25Ti$pval,M1_100Ti$pval,M1_300Ti$pval)

Au_p <- as.data.frame(Au_p)

rownames(Au_p) <- M1_100Ti$geneNames

colnames(Au_p) <- c("25nm","100nm","300nm")

Au <- as.matrix(Au)

ht <- Heatmap(Au,
               cluster_columns = F,
               show_row_names = T,
               name = "Log2FC",
               column_title = "CELLULAR RESPONSE TO METAL IONS",
               column_names_gp = grid::gpar(fontsize = 14),
               column_title_gp = grid::gpar(fontsize = 14, fontface = "bold"),
               colorRamp2(c(-2, 0, 2), c("#2071f5", "white", "#ff4a50")))

Au_p <- as.matrix(Au_p)

ht_p <- Heatmap(Au_p,
                name = "p value",
                show_row_names = T,

```

```

cluster_columns = F,

cluster_rows = F,

width = unit(1, "cm"),

colorRamp2(c(0, 1), c("#3d0c5c", "white")))

ht + ht_p

KEGG_METABOLISM_OF_XENOBIOTICS_BY_CYTOCHROME_P450

msigdbr_3 <- msigdbr(species = "human", category = "C2")

msigdbr_3 <- msigdbr_3[msigdbr_3$gs_subcat == "CP:KEGG",]

msigdbr_3 <- msigdbr_3[msigdbr_3$gs_name ==
"KEGG_METABOLISM_OF_XENOBIOTICS_BY_CYTOCHROME_P450", ]

res25Ti <- read.csv("Results_gene_Ti25nm.csv")

res25Ti <- res25Ti[res25Ti$geneNames != ".",]

M1_25Ti <- res25Ti[res25Ti$geneNames %in% msigdbr_3$gene_symbol,]

M1_25Ti <- M1_25Ti[order(M1_25Ti$geneNames), ]

res100Ti <- read.csv("Results_gene_Ti100nm.csv")

res100Ti <- res100Ti[res100Ti$geneNames != ".",]

M1_100Ti <- res100Ti[res100Ti$geneNames %in% msigdbr_3$gene_symbol,]

M1_100Ti <- M1_100Ti[order(M1_100Ti$geneNames), ]

res300Ti <- read.csv("Results_gene_Ti300nm.csv")

```

```

res300Ti <- res300Ti[res300Ti$geneNames != "."]

M1_300Ti <- res300Ti[res300Ti$geneNames %in% msigdb_r3$gene_symbol,]

M1_300Ti <- M1_300Ti[order(M1_300Ti$geneNames),

M1_100Ti <- M1_100Ti[M1_100Ti$geneNames %in% M1_25Ti$geneNames,]

M1_100Ti <- M1_100Ti[M1_100Ti$geneNames %in% M1_300Ti$geneNames,]

M1_25Ti <- M1_25Ti[M1_25Ti$geneNames %in% M1_100Ti$geneNames,]

M1_300Ti <- M1_300Ti[M1_300Ti$geneNames %in% M1_100Ti$geneNames,]

Au <- cbind(M1_25Ti$logFC,M1_100Ti$logFC,M1_300Ti$logFC)

Au <- as.data.frame(Au)

rownames(Au) <- M1_100Ti$geneNames

colnames(Au) <- c("Ti25nm","Ti100nm","Ti300nm")

Au_p <- cbind(M1_25Ti$pval,M1_100Ti$pval,M1_300Ti$pval)

Au_p <- as.data.frame(Au_p)

rownames(Au_p) <- M1_100Ti$geneNames

colnames(Au_p) <- c("25nm","100nm","300nm")

Au <- as.matrix(Au)

ht <- Heatmap(Au,

               cluster_columns = F,

               show_row_names = T,

               name = "Log2FC",

```

```
column_title = "METABOLISM OF XENOBIOTICS BY CYTOCHROME
P450",
```

```
column_names_gp = grid::gpar(fontsize = 14),
```

```
column_title_gp = grid::gpar(fontsize = 14, fontface = "bold"),
```

```
colorRamp2(c(-2, 0, 2), c("#2071f5", "white", "#ff4a50")))
```

```
Au_p <- as.matrix(Au_p)
```

```
ht_p <- Heatmap(Au_p,
```

```
name = "p value",
```

```
show_row_names = T,
```

```
cluster_columns = F,
```

```
cluster_rows = F,
```

```
width = unit(1, "cm"),
```

```
colorRamp2(c(0, 1), c("#3d0c5c", "white"))))
```

```
ht + ht_p
```

```
HALLMARK_OXIDATIVE_PHOSPHORYLATION
```

```
msigdb_r4 <- msigdb_r4[species = "human",category = "H"]
```

```
msigdb_r4 <- msigdb_r4[msigdb_r4$gs_name ==
"HALLMARK_OXIDATIVE_PHOSPHORYLATION", ]
```

```
res25Ti <- read.csv("Results_gene_Ti25nm.csv")
```

```
res25Ti <- res25Ti[res25Ti$geneNames != ".",]
```

```

M1_25Ti <- res25Ti[res25Ti$geneNames %in% msigdb_r4$gene_symbol,]

M1_25Ti <- M1_25Ti[order(M1_25Ti$geneNames), ]

res100Ti <- read.csv("Results_gene_Ti100nm.csv")

res100Ti <- res100Ti[res100Ti$geneNames != ".",]

M1_100Ti <- res100Ti[res100Ti$geneNames %in% msigdb_r4$gene_symbol,]

M1_100Ti <- M1_100Ti[order(M1_100Ti$geneNames), ]

res300Ti <- read.csv("Results_gene_Ti300nm.csv")

res300Ti <- res300Ti[res300Ti$geneNames != ".",]

M1_300Ti <- res300Ti[res300Ti$geneNames %in% msigdb_r4$gene_symbol,]

M1_300Ti <- M1_300Ti[order(M1_300Ti$geneNames), ]

M1_100Ti <- M1_100Ti[M1_100Ti$geneNames %in% M1_25Ti$geneNames,]

M1_100Ti <- M1_100Ti[M1_100Ti$geneNames %in% M1_300Ti$geneNames,]

M1_25Ti <- M1_25Ti[M1_25Ti$geneNames %in% M1_100Ti$geneNames,]

M1_300Ti <- M1_300Ti[M1_300Ti$geneNames %in% M1_100Ti$geneNames,]

Au <- cbind(M1_25Ti$logFC,M1_100Ti$logFC,M1_300Ti$logFC)

Au <- as.data.frame(Au)

rownames(Au) <- M1_100Ti$geneNames

colnames(Au) <- c("Ti25nm","Ti100nm","Ti300nm")

Au_p <- cbind(M1_25Ti$pval,M1_100Ti$pval,M1_300Ti$pval)

Au_p <- as.data.frame(Au_p)

```

```

rownames(Au_p) <- M1_100Ti$geneNames

colnames(Au_p) <- c("25nm","100nm","300nm")

Au <- as.matrix(Au)

ht <- Heatmap(Au,

  cluster_columns = F,

  show_row_names = T,

  name = "Log2FC",

  column_title = "HALLMARK OXIDATIVE PHOSPHORYLATION",

  column_names_gp = grid::gpar(fontsize = 14),

  column_title_gp = grid::gpar(fontsize = 14, fontface = "bold"),

  colorRamp2(c(-2, 0, 2), c("#2071f5", "white", "#ff4a50")))

Au_p <- as.matrix(Au_p)

ht_p <- Heatmap(Au_p,

  name = "p value",

  show_row_names = T,

  cluster_columns = F,

  cluster_rows = F,

  width = unit(1, "cm"),

  colorRamp2(c(0, 1), c("#3d0c5c", "white")))

ht + ht_p

```

HALLMARK_APOPTOSIS

```
msigdb_5 <- msigdb(species = "human", category = "H")
```

```
msigdb_5 <- msigdb_5[msigdb_5$gs_name == "HALLMARK_APOPTOSIS", ]
```

```
res25Ti <- read.csv("Results_gene_Ti25nm.csv")
```

```
res25Ti <- res25Ti[res25Ti$geneNames != ".",]
```

```
M1_25Ti <- res25Ti[res25Ti$geneNames %in% msigdb_5$gene_symbol,]
```

```
M1_25Ti <- M1_25Ti[order(M1_25Ti$geneNames), ]
```

```
res100Ti <- read.csv("Results_gene_Ti100nm.csv")
```

```
res100Ti <- res100Ti[res100Ti$geneNames != ".",]
```

```
M1_100Ti <- res100Ti[res100Ti$geneNames %in% msigdb_5$gene_symbol,]
```

```
M1_100Ti <- M1_100Ti[order(M1_100Ti$geneNames), ]
```

```
res300Ti <- read.csv("Results_gene_Ti300nm.csv")
```

```
res300Ti <- res300Ti[res300Ti$geneNames != ".",]
```

```
M1_300Ti <- res300Ti[res300Ti$geneNames %in% msigdb_5$gene_symbol,]
```

```
M1_300Ti <- M1_300Ti[order(M1_300Ti$geneNames), ]
```

```
M1_100Ti <- M1_100Ti[M1_100Ti$geneNames %in% M1_25Ti$geneNames,]
```

```
M1_100Ti <- M1_100Ti[M1_100Ti$geneNames %in% M1_300Ti$geneNames,]
```

```
M1_25Ti <- M1_25Ti[M1_25Ti$geneNames %in% M1_100Ti$geneNames,]
```

```
M1_300Ti <- M1_300Ti[M1_300Ti$geneNames %in% M1_100Ti$geneNames,]
```

```
Au <- cbind(M1_25Ti$logFC, M1_100Ti$logFC, M1_300Ti$logFC)
```

```

Au <- as.data.frame(Au)

rownames(Au) <- M1_100Ti$geneNames

colnames(Au) <- c("Ti25nm", "Ti100nm", "Ti300nm")

Au_p <- cbind(M1_25Ti$pval, M1_100Ti$pval, M1_300Ti$pval)

Au_p <- as.data.frame(Au_p)

rownames(Au_p) <- M1_100Ti$geneNames

colnames(Au_p) <- c("25nm", "100nm", "300nm")

Au <- as.matrix(Au)

ht <- Heatmap(Au,

               cluster_columns = F,

               show_row_names = T,

               name = "Log2FC",

               column_title = "HALLMARK APOPTOSIS",

               column_names_gp = grid::gpar(fontsize = 14),

               column_title_gp = grid::gpar(fontsize = 14, fontface = "bold"),

               colorRamp2(c(-2, 0, 2), c("#2071f5", "white", "#ff4a50")))

Au_p <- as.matrix(Au_p)

ht_p <- Heatmap(Au_p,

                name = "p value",

                show_row_names = T,

```

```

cluster_columns = F,

cluster_rows = F,

width = unit(1, "cm"),

colorRamp2(c(0, 1), c("#3d0c5c", "white")))

ht + ht_p

ALONSO_METASTASIS_EMT

msigdbr_6 <- msigdbr(species = "human",category = "C2")

msigdbr_6 <- msigdbr_6[msigdbr_6$gs_subcat == "CGP",]

msigdbr_6 <- msigdbr_6[msigdbr_6$gs_name ==
"ALONSO_METASTASIS_EMT_UP", ]

msigdbr_6_ <- msigdbr(species = "human",category = "C2")

msigdbr_6_ <- msigdbr_6_[msigdbr_6_$gs_subcat == "CGP",]

msigdbr_6_ <- msigdbr_6_[msigdbr_6_$gs_name ==
"ALONSO_METASTASIS_EMT_DN", ]

msigdbr_6 <- rbind(msigdbr_6,msigdbr_6_)

res25Ti <- read.csv("Results_gene_Ti25nm.csv")

res25Ti <- res25Ti[res25Ti$geneNames != ".",]

M1_25Ti <- res25Ti[res25Ti$geneNames %in% msigdbr_6$gene_symbol,]

M1_25Ti <- M1_25Ti[order(M1_25Ti$geneNames), ]

res100Ti <- read.csv("Results_gene_Ti100nm.csv")

res100Ti <- res100Ti[res100Ti$geneNames != ".",]

```

```

M1_100Ti <- res100Ti[res100Ti$geneNames %in% msigdb6$gene_symbol,]

M1_100Ti <- M1_100Ti[order(M1_100Ti$geneNames), ]

res300Ti <- read.csv("Results_gene_Ti300nm.csv")

res300Ti <- res300Ti[res300Ti$geneNames != ".",]

M1_300Ti <- res300Ti[res300Ti$geneNames %in% msigdb6$gene_symbol,]

M1_300Ti <- M1_300Ti[order(M1_300Ti$geneNames), ]

M1_100Ti <- M1_100Ti[M1_100Ti$geneNames %in% M1_25Ti$geneNames,]

M1_100Ti <- M1_100Ti[M1_100Ti$geneNames %in% M1_300Ti$geneNames,]

M1_25Ti <- M1_25Ti[M1_25Ti$geneNames %in% M1_100Ti$geneNames,]

M1_300Ti <- M1_300Ti[M1_300Ti$geneNames %in% M1_100Ti$geneNames,]

Au <- cbind(M1_25Ti$logFC,M1_100Ti$logFC,M1_300Ti$logFC)

Au <- as.data.frame(Au)

rownames(Au) <- M1_100Ti$geneNames

colnames(Au) <- c("Ti25nm","Ti100nm","Ti300nm")

Au_p <- cbind(M1_25Ti$pval,M1_100Ti$pval,M1_300Ti$pval)

Au_p <- as.data.frame(Au_p)

rownames(Au_p) <- M1_100Ti$geneNames

colnames(Au_p) <- c("25nm","100nm","300nm")

Au <- as.matrix(Au)

ht <- Heatmap(Au,

```

```

cluster_columns = F,

show_row_names = T,

name = "Log2FC",

column_title = "METASTASIS EMT",

column_names_gp = grid::gpar(fontsize = 14),

column_title_gp = grid::gpar(fontsize = 14, fontface = "bold"),

colorRamp2(c(-2, 0, 2), c("#2071f5", "white", "#ff4a50")))

Au_p <- as.matrix(Au_p)

ht_p <- Heatmap(Au_p,

  name = "p value",

  show_row_names = T,

  cluster_columns = F,

  cluster_rows = F,

  width = unit(1, "cm"),

  colorRamp2(c(0, 1), c("#3d0c5c", "white")))

ht + ht_p

LIVER CANCER

msigdb_r7_2 <- msigdb_r7_2[msigdb_r7_2$gs_subcat == "CGP",]

```

```
msigdbr_7_2 <- msigdbr_7_2[msigdbr_7_2$gs_name == "PATIL_LIVER_CANCER",  
]
```

```
msigdbr_7_3 <- msigdbr(species = "human",category = "C2")
```

```
msigdbr_7_3 <- msigdbr_7_3[msigdbr_7_3$gs_subcat == "CGP",]
```

```
msigdbr_7_3 <- msigdbr_7_3[msigdbr_7_3$gs_name == "SMITH_LIVER_CANCER",  
]
```

```
msigdbr_7 <- rbind(msigdbr_7_2,msigdbr_7_3)
```

```
msigdbr_7 <- msigdbr_7[!duplicated(msigdbr_7$gene_symbol), ]
```

```
res25Ti <- read.csv("Results_gene_Ti25nm.csv")
```

```
res25Ti <- res25Ti[res25Ti$geneNames != ".",]
```

```
M1_25Ti <- res25Ti[res25Ti$geneNames %in% msigdbr_7$gene_symbol,]
```

```
M1_25Ti <- M1_25Ti[order(M1_25Ti$geneNames), ]
```

```
M1_25Ti <- M1_25Ti %>%
```

```
  group_by(geneNames) %>%
```

```
  summarise(logFC = mean(logFC), pval = first(pval))
```

```
res100Ti <- read.csv("Results_gene_Ti100nm.csv")
```

```
res100Ti <- res100Ti[res100Ti$geneNames != ".",]
```

```
M1_100Ti <- res100Ti[res100Ti$geneNames %in% msigdbr_7$gene_symbol,]
```

```
M1_100Ti <- M1_100Ti[order(M1_100Ti$geneNames), ]
```

```
M1_100Ti <- M1_100Ti %>%
```

```

group_by(geneNames) %>%

summarise(logFC = mean(logFC), pval = first(pval))

res300Ti <- read.csv("Results_gene_Ti300nm.csv")

res300Ti <- res300Ti[res300Ti$geneNames != ".",]

M1_300Ti <- res300Ti[res300Ti$geneNames %in% msigdb_r7$gene_symbol,]

M1_300Ti <- M1_300Ti[order(M1_300Ti$geneNames), ]

M1_300Ti <- M1_300Ti %>%

group_by(geneNames) %>%

summarise(logFC = mean(logFC), pval = first(pval))

M1_25Ti <- M1_25Ti[M1_25Ti$geneNames %in% M1_300Ti$geneNames,]

M1_25Ti <- M1_25Ti[M1_25Ti$geneNames %in% M1_100Ti$geneNames,]

M1_100Ti <- M1_100Ti[M1_100Ti$geneNames %in% M1_25Ti$geneNames,]

M1_300Ti <- M1_300Ti[M1_300Ti$geneNames %in% M1_25Ti$geneNames,]

Au <- cbind(M1_25Ti$logFC,M1_100Ti$logFC,M1_300Ti$logFC)

Au <- as.data.frame(Au)

rownames(Au) <- M1_100Ti$geneNames

colnames(Au) <- c("Ti25nm","Ti100nm","Ti300nm")

Au_p <- cbind(M1_25Ti$pval,M1_100Ti$pval,M1_300Ti$pval)

Au_p <- as.data.frame(Au_p)

rownames(Au_p) <- M1_100Ti$geneNames

```

```

colnames(Au_p) <- c("25nm","100nm","300nm")

Au <- as.matrix(Au)

ht <- Heatmap(Au,

               cluster_columns = F,

               show_row_names = F,

               name = "Log2FC",

               column_title = "LIVER CANCER",

               column_names_gp = grid::gpar(fontsize = 14),

               column_title_gp = grid::gpar(fontsize = 14, fontface = "bold"),

               colorRamp2(c(-2, 0, 2), c("#2071f5", "white", "#ff4a50")))

Au_p <- as.matrix(Au_p)

ht_p <- Heatmap(Au_p,

                name = "p value",

                show_row_names = F,

                cluster_columns = F,

                cluster_rows = F,

                width = unit(1, "cm"),

                colorRamp2(c(0, 1), c("#3d0c5c", "white")))

ht + ht_p

```

HALLMARK_REACTIVE_OXYGEN_SPECIES_PATHWAY

```

msigdb_10 <- msigdb(species = "human",category = "H")

msigdb_10 <- msigdb_10[msigdb_10$gs_name ==
"HALLMARK_REACTIVE_OXYGEN_SPECIES_PATHWAY", ]

res25Ti <- read.csv("Results_gene_Ti25nm.csv")

res25Ti <- res25Ti[res25Ti$geneNames != ".",]

M1_25Ti <- res25Ti[res25Ti$geneNames %in% msigdb_10$gene_symbol,]

M1_25Ti <- M1_25Ti[order(M1_25Ti$geneNames), ]

res100Ti <- read.csv("Results_gene_Ti100nm.csv")

res100Ti <- res100Ti[res100Ti$geneNames != ".",]

M1_100Ti <- res100Ti[res100Ti$geneNames %in% msigdb_10$gene_symbol,]

M1_100Ti <- M1_100Ti[order(M1_100Ti$geneNames), ]

res300Ti <- read.csv("Results_gene_Ti300nm.csv")

res300Ti <- res300Ti[res300Ti$geneNames != ".",]

M1_300Ti <- res300Ti[res300Ti$geneNames %in% msigdb_10$gene_symbol,]

M1_300Ti <- M1_300Ti[order(M1_300Ti$geneNames), ]

M1_100Ti <- M1_100Ti[M1_100Ti$geneNames %in% M1_25Ti$geneNames,]

M1_100Ti <- M1_100Ti[M1_100Ti$geneNames %in% M1_300Ti$geneNames,]

M1_25Ti <- M1_25Ti[M1_25Ti$geneNames %in% M1_100Ti$geneNames,]

M1_300Ti <- M1_300Ti[M1_300Ti$geneNames %in% M1_100Ti$geneNames,]

Au <- cbind(M1_25Ti$logFC,M1_100Ti$logFC,M1_300Ti$logFC)

```

```

Au <- as.data.frame(Au)

rownames(Au) <- M1_100Ti$geneNames

colnames(Au) <- c("Ti25nm", "Ti100nm", "Ti300nm")

Au_p <- cbind(M1_25Ti$pval, M1_100Ti$pval, M1_300Ti$pval)

Au_p <- as.data.frame(Au_p)

rownames(Au_p) <- M1_100Ti$geneNames

colnames(Au_p) <- c("25nm", "100nm", "300nm")

Au <- as.matrix(Au)

ht <- Heatmap(Au,

               cluster_columns = F,

               show_row_names = T,

               name = "Log2FC",

               column_title = "HALLMARK ROS PATHWAY",

               column_names_gp = grid::gpar(fontsize = 14),

               column_title_gp = grid::gpar(fontsize = 14, fontface = "bold"),

               colorRamp2(c(-2, 0, 2), c("#2071f5", "white", "#ff4a50")))

Au_p <- as.matrix(Au_p)

ht_p <- Heatmap(Au_p,

                name = "p value",

                show_row_names = T,

```

```

cluster_columns = F,

cluster_rows = F,

width = unit(1, "cm"),

colorRamp2(c(0, 1), c("#3d0c5c", "white")))

ht + ht_p

```

HEATMAPS OF QUANTUM DOTS

REACTOME_METAL_ION_SLC_TRANSPORTERS

```

msigdbr_1 <- msigdbr(species = "human",category = "C2")

msigdbr_1 <- msigdbr_1[msigdbr_1$gs_subcat == "CP:REACTOME",]

msigdbr_1 <- msigdbr_1[msigdbr_1$gs_name ==
"REACTOME_METAL_ION_SLC_TRANSPORTERS", ]

resQD1 <- read.csv("Results_gene_QD1.csv")

resQD1 <- resQD1[resQD1$geneNames != ".",]

M1_QD1 <- resQD1[resQD1$geneNames %in% msigdbr_1$gene_symbol,]

M1_QD1 <- M1_QD1[order(M1_QD1$geneNames), ]

resQD2 <- read.csv("Results_gene_QD2.csv")

resQD2 <- resQD2[resQD2$geneNames != ".",]

M1_QD2 <- resQD2[resQD2$geneNames %in% msigdbr_1$gene_symbol,]

M1_QD2 <- M1_QD2[order(M1_QD2$geneNames), ]

```

```

M1_QD2 <- M1_QD2[M1_QD2$geneNames %in% M1_QD1$geneNames,]

M1_QD1 <- M1_QD1[M1_QD1$geneNames %in% M1_QD2$geneNames,]

Au <- cbind(M1_QD1$logFC,M1_QD2$logFC)

Au <- as.data.frame(Au)

rownames(Au) <- M1_QD2$geneNames

colnames(Au) <- c("QD1","QD2")

Au_p <- cbind(M1_QD1$pval,M1_QD2$pval)

Au_p <- as.data.frame(Au_p)

rownames(Au_p) <- M1_QD2$geneNames

colnames(Au_p) <- c("QD1","QD2")

Au <- as.matrix(Au)

ht <- Heatmap(Au,

               cluster_columns = F,

               show_row_names = T,

               name = "Log2FC",

               column_title = "METAL ION SLC TRANSPORTERS",

               column_names_gp = grid::gpar(fontsize = 14),

               column_title_gp = grid::gpar(fontsize = 14, fontface = "bold"),

               colorRamp2(c(-2, 0, 2), c("#2071f5", "white", "#ff4a50")))

Au_p <- as.matrix(Au_p)

```

```

ht_p <- Heatmap(Au_p,

                name = "p value",

                show_row_names = T,

                cluster_columns = F,

                cluster_rows = F,

                width = unit(1, "cm"),

                colorRamp2(c(0, 1), c("#3d0c5c", "white")))

ht + ht_p

REACTOME_RESPONSE_TO_METAL_IONS

msigdb_r2 <- msigdb_r2[species == "human",category == "C2"]

msigdb_r2 <- msigdb_r2[msigdb_r2$gs_subcat == "CP:REACTOME",]

msigdb_r2 <- msigdb_r2[msigdb_r2$gs_name ==
"REACTOME_RESPONSE_TO_METAL_IONS", ]

resQD1 <- read.csv("Results_gene_QD1.csv")

resQD1 <- resQD1[resQD1$geneNames != ".",]

M1_QD1 <- resQD1[resQD1$geneNames %in% msigdb_r2$gene_symbol,]

M1_QD1 <- M1_QD1[order(M1_QD1$geneNames), ]

resQD2 <- read.csv("Results_gene_QD2.csv")

resQD2 <- resQD2[resQD2$geneNames != ".",]

M1_QD2 <- resQD2[resQD2$geneNames %in% msigdb_r2$gene_symbol,]

```

```

M1_QD2 <- M1_QD2[order(M1_QD2$geneNames), ]

M1_QD2 <- M1_QD2[M1_QD2$geneNames %in% M1_QD1$geneNames,]

M1_QD1 <- M1_QD1[M1_QD1$geneNames %in% M1_QD2$geneNames,]

Au <- cbind(M1_QD1$logFC,M1_QD2$logFC)

Au <- as.data.frame(Au)

rownames(Au) <- M1_QD2$geneNames

colnames(Au) <- c("QD1","QD2")

Au_p <- cbind(M1_QD1$pval,M1_QD2$pval)

Au_p <- as.data.frame(Au_p)

rownames(Au_p) <- M1_QD2$geneNames

colnames(Au_p) <- c("QD1","QD2")

Au <- as.matrix(Au)

ht <- Heatmap(Au,

               cluster_columns = F,

               show_row_names = T,

               name = "Log2FC",

               column_title = "CELLULAR RESPONSE TO METAL IONS",

               column_names_gp = grid::gpar(fontsize = 14),

               column_title_gp = grid::gpar(fontsize = 14, fontface = "bold"),

               colorRamp2(c(-2, 0, 2), c("#2071f5", "white", "#ff4a50")))

```

```

Au_p <- as.matrix(Au_p)

ht_p <- Heatmap(Au_p,

  name = "p value",

  show_row_names = T,

  cluster_columns = F,

  cluster_rows = F,

  width = unit(1, "cm"),

  colorRamp2(c(0, 1), c("#3d0c5c", "white")))

ht + ht_p

KEGG_METABOLISM_OF_XENOBIOTICS_BY_CYTOCHROME_P450

msigdb_r3 <- msigdb_r3[species == "human",category == "C2"]

msigdb_r3 <- msigdb_r3[msigdb_r3$gs_subcat == "CP:KEGG",]

msigdb_r3 <- msigdb_r3[msigdb_r3$gs_name ==
"KEGG_METABOLISM_OF_XENOBIOTICS_BY_CYTOCHROME_P450", ]

resQD1 <- read.csv("Results_gene_QD1.csv")

resQD1 <- resQD1[resQD1$geneNames != ".",]

M1_QD1 <- resQD1[resQD1$geneNames %in% msigdb_r3$gene_symbol,]

M1_QD1 <- M1_QD1[order(M1_QD1$geneNames), ]

resQD2 <- read.csv("Results_gene_QD2.csv")

resQD2 <- resQD2[resQD2$geneNames != ".",]

```

```

M1_QD2 <- resQD2[resQD2$geneNames %in% msigdb_r3$gene_symbol,]

M1_QD2 <- M1_QD2[order(M1_QD2$geneNames), ]

M1_QD2 <- M1_QD2[M1_QD2$geneNames %in% M1_QD1$geneNames,]

M1_QD1 <- M1_QD1[M1_QD1$geneNames %in% M1_QD2$geneNames,]

Au <- cbind(M1_QD1$logFC,M1_QD2$logFC)

Au <- as.data.frame(Au)

rownames(Au) <- M1_QD2$geneNames

colnames(Au) <- c("QD1","QD2")

Au_p <- cbind(M1_QD1$pval,M1_QD2$pval)

Au_p <- as.data.frame(Au_p)

rownames(Au_p) <- M1_QD2$geneNames

colnames(Au_p) <- c("QD1","QD2")

Au <- as.matrix(Au)

ht <- Heatmap(Au,

               cluster_columns = F,

               show_row_names = T,

               name = "Log2FC",

               column_title = "METABOLISM OF XENOBIOTICS BY CYTOCHROME
P450",

               column_names_gp = grid::gpar(fontsize = 14),

```

```

column_title_gp = grid::gpar(fontsize = 14, fontface = "bold"),

colorRamp2(c(-2, 0, 2), c("#2071f5", "white", "#ff4a50")))

Au_p <- as.matrix(Au_p)

ht_p <- Heatmap(Au_p,

  name = "p value",

  show_row_names = T,

  cluster_columns = F,

  cluster_rows = F,

  width = unit(1, "cm"),

  colorRamp2(c(0, 1), c("#3d0c5c", "white")))

ht + ht_p

```

HALLMARK_OXIDATIVE_PHOSPHORYLATIO

```
msigdbr_4 <- msigdbr(species = "human",category = "H")
```

```
msigdbr_4 <- msigdbr_4[msigdbr_4$gs_name ==
"HALLMARK_OXIDATIVE_PHOSPHORYLATION", ]
```

```
resQD1 <- read.csv("Results_gene_QD1.csv")
```

```
resQD1 <- resQD1[resQD1$geneNames != ".",]
```

```
M1_QD1 <- resQD1[resQD1$geneNames %in% msigdbr_4$gene_symbol,]
```

```
M1_QD1 <- M1_QD1[order(M1_QD1$geneNames), ]
```

```
resQD2 <- read.csv("Results_gene_QD2.csv")
```

```

resQD2 <- resQD2[resQD2$geneNames != ".",]

M1_QD2 <- resQD2[resQD2$geneNames %in% msigdb_r4$gene_symbol,]

M1_QD2 <- M1_QD2[order(M1_QD2$geneNames), ]

M1_QD2 <- M1_QD2[M1_QD2$geneNames %in% M1_QD1$geneNames,]

M1_QD1 <- M1_QD1[M1_QD1$geneNames %in% M1_QD2$geneNames,]


Au <- cbind(M1_QD1$logFC,M1_QD2$logFC)

Au <- as.data.frame(Au)

rownames(Au) <- M1_QD2$geneNames

colnames(Au) <- c("QD1","QD2")

Au_p <- cbind(M1_QD1$pval,M1_QD2$pval)

Au_p <- as.data.frame(Au_p)

rownames(Au_p) <- M1_QD2$geneNames

colnames(Au_p) <- c("QD1","QD2")

Au <- as.matrix(Au)

ht <- Heatmap(Au,

               cluster_columns = F,

               show_row_names = T,

               name = "Log2FC",

               column_title = "HALLMARK OXIDATIVE PHOSPHORYLATION",

```

```

column_names_gp = grid::gpar(fontsize = 14),

column_title_gp = grid::gpar(fontsize = 14, fontface = "bold"),

colorRamp2(c(-2, 0, 2), c("#2071f5", "white", "#ff4a50")))

Au_p <- as.matrix(Au_p)

ht_p <- Heatmap(Au_p,

               name = "p value",

               show_row_names = T,

               cluster_columns = F,

               cluster_rows = F,

               width = unit(1, "cm"),

               colorRamp2(c(0, 1), c("#3d0c5c", "white")))

ht + ht_p

```

HALLMARK_APOPTOSIS

```

msigdb_r5 <- msigdb_r5[species == "human",category == "H"]

msigdb_r5 <- msigdb_r5[msigdb_r5$gs_name == "HALLMARK_APOPTOSIS", ]

resQD1 <- read.csv("Results_gene_QD1.csv")

resQD1 <- resQD1[resQD1$geneNames != ".",]

M1_QD1 <- resQD1[resQD1$geneNames %in% msigdb_r5$gene_symbol,]

M1_QD1 <- M1_QD1[order(M1_QD1$geneNames), ]

resQD2 <- read.csv("Results_gene_QD2.csv")

```

```

resQD2 <- resQD2[resQD2$geneNames != ".",]

M1_QD2 <- resQD2[resQD2$geneNames %in% msigdb_r5$gene_symbol,]

M1_QD2 <- M1_QD2[order(M1_QD2$geneNames), ]

M1_QD2 <- M1_QD2[M1_QD2$geneNames %in% M1_QD1$geneNames,]

M1_QD1 <- M1_QD1[M1_QD1$geneNames %in% M1_QD2$geneNames,]


Au <- cbind(M1_QD1$logFC,M1_QD2$logFC)

Au <- as.data.frame(Au)

rownames(Au) <- M1_QD2$geneNames

colnames(Au) <- c("QD1","QD2")

Au_p <- cbind(M1_QD1$pval,M1_QD2$pval)

Au_p <- as.data.frame(Au_p)

rownames(Au_p) <- M1_QD2$geneNames

colnames(Au_p) <- c("QD1","QD2")

Au <- as.matrix(Au)

ht <- Heatmap(Au,

               cluster_columns = F,

               show_row_names = T,

               name = "Log2FC",

               column_title = "HALLMARK APOPTOSIS",

```

```

column_names_gp = grid::gpar(fontsize = 14),

column_title_gp = grid::gpar(fontsize = 14, fontface = "bold"),

colorRamp2(c(-2, 0, 2), c("#2071f5", "white", "#ff4a50")))

Au_p <- as.matrix(Au_p)

ht_p <- Heatmap(Au_p,

  name = "p value",

  show_row_names = T,

  cluster_columns = F,

  cluster_rows = F,

  width = unit(1, "cm"),

  colorRamp2(c(0, 1), c("#3d0c5c", "white")))

ht + ht_p

```

ALONSO_METASTASIS_EMT

```
msigdbr_6 <- msigdbr(species = "human",category = "C2")
```

```
msigdbr_6 <- msigdbr_6[msigdbr_6$gs_subcat == "CGP",]
```

```
msigdbr_6 <- msigdbr_6[msigdbr_6$gs_name ==
"ALONSO_METASTASIS_EMT_UP", ]
```

```
msigdbr_6_ <- msigdbr(species = "human",category = "C2")
```

```
msigdbr_6_ <- msigdbr_6_[msigdbr_6_$gs_subcat == "CGP",]
```

```
msigdbr_6_ <- msigdbr_6_[msigdbr_6_$gs_name ==
"ALONSO_METASTASIS_EMT_DN", ]
```

```

msigdbr_6 <- rbind(msigdbr_6,msigdbr_6_)

resQD1 <- read.csv("Results_gene_QD1.csv")

resQD1 <- resQD1[resQD1$geneNames != ".",]

M1_QD1 <- resQD1[resQD1$geneNames %in% msigdbr_6$gene_symbol,]

M1_QD1 <- M1_QD1[order(M1_QD1$geneNames), ]

resQD2 <- read.csv("Results_gene_QD2.csv")

resQD2 <- resQD2[resQD2$geneNames != ".",]

M1_QD2 <- resQD2[resQD2$geneNames %in% msigdbr_6$gene_symbol,]

M1_QD2 <- M1_QD2[order(M1_QD2$geneNames), ]

M1_QD2 <- M1_QD2[M1_QD2$geneNames %in% M1_QD1$geneNames,]

M1_QD1 <- M1_QD1[M1_QD1$geneNames %in% M1_QD2$geneNames,]

Au <- cbind(M1_QD1$logFC,M1_QD2$logFC)

Au <- as.data.frame(Au)

rownames(Au) <- M1_QD2$geneNames

colnames(Au) <- c("QD1","QD2")

Au_p <- cbind(M1_QD1$pval,M1_QD2$pval)

Au_p <- as.data.frame(Au_p)

rownames(Au_p) <- M1_QD2$geneNames

colnames(Au_p) <- c("QD1","QD2")

```

```

Au <- as.matrix(Au)

ht <- Heatmap(Au,

  cluster_columns = F,

  show_row_names = T,

  name = "Log2FC",

  column_title = "METASTASIS EMT",

  column_names_gp = grid::gpar(fontsize = 14),

  column_title_gp = grid::gpar(fontsize = 14, fontface = "bold"),

  colorRamp2(c(-2, 0, 2), c("#2071f5", "white", "#ff4a50")))

Au_p <- as.matrix(Au_p)

ht_p <- Heatmap(Au_p,

  name = "p value",

  show_row_names = T,

  cluster_columns = F,

  cluster_rows = F,

  width = unit(1, "cm"),

  colorRamp2(c(0, 1), c("#3d0c5c", "white")))

ht + ht_p

LIVER CANCER

msigdb_r7_2 <- msigdb(species = "human", category = "C2")

```

```

msigdbr_7_2 <- msigdbr_7_2[msigdbr_7_2$gs_subcat == "CGP",]

msigdbr_7_2 <- msigdbr_7_2[msigdbr_7_2$gs_name == "PATIL_LIVER_CANCER",
]

msigdbr_7_3 <- msigdbr(species = "human",category = "C2")

msigdbr_7_3 <- msigdbr_7_3[msigdbr_7_3$gs_subcat == "CGP",]

msigdbr_7_3 <- msigdbr_7_3[msigdbr_7_3$gs_name == "SMITH_LIVER_CANCER",
]

msigdbr_7 <- rbind(msigdbr_7_2,msigdbr_7_3)

msigdbr_7 <- msigdbr_7[!duplicated(msigdbr_7$gene_symbol), ]

resQD1 <- read.csv("Results_gene_QD1.csv")

resQD1 <- resQD1[resQD1$geneNames != ".",]

M1_QD1 <- resQD1[resQD1$geneNames %in% msigdbr_7$gene_symbol,]

M1_QD1 <- M1_QD1[order(M1_QD1$geneNames), ]

M1_QD1 <- M1_QD1 %>%

  group_by(geneNames) %>%

  summarise(logFC = mean(logFC), pval = first(pval))

resQD2 <- read.csv("Results_gene_QD2.csv")

resQD2 <- resQD2[resQD2$geneNames != ".",]

M1_QD2 <- resQD2[resQD2$geneNames %in% msigdbr_7$gene_symbol,]

M1_QD2 <- M1_QD2[order(M1_QD2$geneNames), ]

M1_QD2 <- M1_QD2 %>%

```

```

group_by(geneNames) %>%

summarise(logFC = mean(logFC), pval = first(pval))

M1_QD1 <- M1_QD1[M1_QD1$geneNames %in% M1_QD2$geneNames,]

M1_QD2 <- M1_QD2[M1_QD2$geneNames %in% M1_QD1$geneNames,]

Au <- cbind(M1_QD1$logFC,M1_QD2$logFC)

Au <- as.data.frame(Au)

rownames(Au) <- M1_QD2$geneNames

colnames(Au) <- c("QD1","QD2")


Au_p <- cbind(M1_QD1$pval,M1_QD2$pval)

Au_p <- as.data.frame(Au_p)

rownames(Au_p) <- M1_QD2$geneNames

colnames(Au_p) <- c("QD1","QD2")

Au <- as.matrix(Au)

ht <- Heatmap(Au,

               cluster_columns = F,

               show_row_names = F,

               name = "Log2FC",

               column_title = "LIVER CANCER",

               column_names_gp = grid::gpar(fontsize = 14),

```

```

column_title_gp = grid::gpar(fontsize = 14, fontface = "bold"),

colorRamp2(c(-2, 0, 2), c("#2071f5", "white", "#ff4a50")))

Au_p <- as.matrix(Au_p)

ht_p <- Heatmap(Au_p,

  name = "p value",

  show_row_names = F,

  cluster_columns = F,

  cluster_rows = F,

  width = unit(1, "cm"),

  colorRamp2(c(0, 1), c("#3d0c5c", "white")))

ht + ht_p

LIVER_CANCER_UP

msigdbr_8 <- msigdbr(species = "human",category = "C2")

msigdbr_8 <- msigdbr_8[msigdbr_8$gs_subcat == "CGP",]

msigdbr_8 <- msigdbr_8[msigdbr_8$gs_name ==
"ACEVEDO_LIVER_CANCER_UP", ]

resQD1 <- read.csv("Results_gene_QD1.csv")

resQD1 <- resQD1[resQD1$geneNames != ".",]

M1_QD1 <- resQD1[resQD1$geneNames %in% msigdbr_8$gene_symbol,]

M1_QD1 <- M1_QD1[order(M1_QD1$geneNames), ]

```

```

M1_QD1 <- M1_QD1 %>%

  group_by(geneNames) %>%

  summarise(logFC = mean(logFC), pval = first(pval))

resQD2 <- read.csv("Results_gene_QD2.csv")

resQD2 <- resQD2[resQD2$geneNames != ".",]

M1_QD2 <- resQD2[resQD2$geneNames %in% msigdb_r8$gene_symbol,]

M1_QD2 <- M1_QD2[order(M1_QD2$geneNames), ]

M1_QD2 <- M1_QD2 %>%

  group_by(geneNames) %>%

  summarise(logFC = mean(logFC), pval = first(pval))

M1_QD1 <- M1_QD1[M1_QD1$geneNames %in% M1_QD2$geneNames,]

M1_QD2 <- M1_QD2[M1_QD2$geneNames %in% M1_QD1$geneNames,]

Au <- cbind(M1_QD1$logFC,M1_QD2$logFC)

Au <- as.data.frame(Au)

rownames(Au) <- M1_QD2$geneNames

colnames(Au) <- c("QD1","QD2")

Au_p <- cbind(M1_QD1$pval,M1_QD2$pval)

Au_p <- as.data.frame(Au_p)

rownames(Au_p) <- M1_QD2$geneNames

colnames(Au_p) <- c("QD1","QD2")

```

```

Au <- as.matrix(Au)

ht <- Heatmap(Au,

  cluster_columns = F,

  show_row_names = F,

  name = "Log2FC",

  column_title = "LIVER CANCER UP",

  column_names_gp = grid::gpar(fontsize = 14),

  column_title_gp = grid::gpar(fontsize = 14, fontface = "bold"),

  colorRamp2(c(-2, 0, 2), c("#2071f5", "white", "#ff4a50")))

Au_p <- as.matrix(Au_p)

ht_p <- Heatmap(Au_p,

  name = "p value",

  show_row_names = F,

  cluster_columns = F,

  cluster_rows = F,

  width = unit(1, "cm"),

  colorRamp2(c(0, 1), c("#3d0c5c", "white")))

ht + ht_p

LIVER_CANCER_DN

msigdb_r_9 <- msigdb_r(species = "human", category = "C2")

```

```

msigdb_9 <- msigdb_9[msigdb_9$gs_subcat == "CGP",]

msigdb_9 <- msigdb_9[msigdb_9$gs_name ==
"ACEVEDO_LIVER_CANCER_DN", ]

resQD1 <- read.csv("Results_gene_QD1.csv")

resQD1 <- resQD1[resQD1$geneNames != ".",]

M1_QD1 <- resQD1[resQD1$geneNames %in% msigdb_9$gene_symbol,]

M1_QD1 <- M1_QD1[order(M1_QD1$geneNames), ]

M1_QD1 <- M1_QD1 %>%

  group_by(geneNames) %>%

  summarise(logFC = mean(logFC), pval = first(pval))

resQD2 <- read.csv("Results_gene_QD2.csv")

resQD2 <- resQD2[resQD2$geneNames != ".",]

M1_QD2 <- resQD2[resQD2$geneNames %in% msigdb_9$gene_symbol,]

M1_QD2 <- M1_QD2[order(M1_QD2$geneNames), ]

M1_QD2 <- M1_QD2 %>%

  group_by(geneNames) %>%

  summarise(logFC = mean(logFC), pval = first(pval))

M1_QD1 <- M1_QD1[M1_QD1$geneNames %in% M1_QD2$geneNames,]

M1_QD2 <- M1_QD2[M1_QD2$geneNames %in% M1_QD1$geneNames,]

Au <- cbind(M1_QD1$logFC,M1_QD2$logFC)

```

```

Au <- as.data.frame(Au)

rownames(Au) <- M1_QD2$geneNames

colnames(Au) <- c("QD1","QD2")

Au_p <- cbind(M1_QD1$pval,M1_QD2$pval)

Au_p <- as.data.frame(Au_p)

rownames(Au_p) <- M1_QD2$geneNames

colnames(Au_p) <- c("QD1","QD2")

Au <- as.matrix(Au)

ht <- Heatmap(Au,

               cluster_columns = F,

               show_row_names = F,

               name = "Log2FC",

               column_title = "LIVER CANCER DN",

               column_names_gp = grid::gpar(fontsize = 14),

               column_title_gp = grid::gpar(fontsize = 14, fontface = "bold"),

               colorRamp2(c(-2, 0, 2), c("#2071f5", "white", "#ff4a50")))

Au_p <- as.matrix(Au_p)

ht_p <- Heatmap(Au_p,

                name = "p value",

                show_row_names = F,

```

```

cluster_columns = F,

cluster_rows = F,

width = unit(1, "cm"),

colorRamp2(c(0, 1), c("#3d0c5c", "white")))

ht + ht_p

```

```
#####
```

```
HALLMARK_REACTIVE_OXYGEN_SPECIES_PATHWAY
```

```
msigdbr_10 <- msigdbr(species = "human",category = "H")
```

```
msigdbr_10 <- msigdbr_10[msigdbr_10$gs_name ==
"HALLMARK_REACTIVE_OXYGEN_SPECIES_PATHWAY", ]
```

```
resQD1 <- read.csv("Results_gene_QD1.csv")
```

```
resQD1 <- resQD1[resQD1$geneNames != ".",]
```

```
M1_QD1 <- resQD1[resQD1$geneNames %in% msigdbr_10$gene_symbol,]
```

```
M1_QD1 <- M1_QD1[order(M1_QD1$geneNames), ]
```

```
resQD2 <- read.csv("Results_gene_QD2.csv")
```

```
resQD2 <- resQD2[resQD2$geneNames != ".",]
```

```
M1_QD2 <- resQD2[resQD2$geneNames %in% msigdbr_10$gene_symbol,]
```

```
M1_QD2 <- M1_QD2[order(M1_QD2$geneNames), ]
```

```

M1_QD2 <- M1_QD2[M1_QD2$geneNames %in% M1_QD1$geneNames,]

M1_QD1 <- M1_QD1[M1_QD1$geneNames %in% M1_QD2$geneNames,]

Au <- cbind(M1_QD1$logFC,M1_QD2$logFC)

Au <- as.data.frame(Au)

rownames(Au) <- M1_QD2$geneNames

colnames(Au) <- c("QD1","QD2")

Au_p <- cbind(M1_QD1$pval,M1_QD2$pval)

Au_p <- as.data.frame(Au_p)

rownames(Au_p) <- M1_QD2$geneNames

colnames(Au_p) <- c("QD1","QD2")

Au <- as.matrix(Au)

ht <- Heatmap(Au,

               cluster_columns = F,

               show_row_names = T,

               name = "Log2FC",

               column_title = "HALLMARK ROS PATHWAY",

               column_names_gp = grid::gpar(fontsize = 14),

               column_title_gp = grid::gpar(fontsize = 14, fontface = "bold"),

               colorRamp2(c(-2, 0, 2), c("#2071f5", "white", "#ff4a50")))

Au_p <- as.matrix(Au_p)

```

```

ht_p <- Heatmap(Au_p,

                name = "p value",

                show_row_names = T,

                cluster_columns = F,

                cluster_rows = F,

                width = unit(1, "cm"),

                colorRamp2(c(0, 1), c("#3d0c5c", "white")))

ht + ht_p

Significant Gene Lists

resQD1 <- read.csv("Results_gene_QD1.csv")

resQD1 <- resQD1[resQD1$geneNames != ".",]

resQD1 <- resQD1[resQD1$pval < 0.05,]

resQD2 <- read.csv("Results_gene_QD2.csv")

resQD2 <- resQD2[resQD2$geneNames != ".",]

resQD2 <- resQD2[resQD2$pval < 0.05,]

resQD1 <- resQD1[resQD1$logFC >= 0.5 | resQD1$logFC <= -0.5, ]

resQD2 <- resQD2[resQD2$logFC >= 0.5 | resQD2$logFC <= -0.5, ]

gene_list <- bitr(resQD2$geneNames, fromType="SYMBOL", toType="ENTREZID",
OrgDb="org.Hs.eg.db")

univ <- Lkeys(org.Hs.egGO)

```

```
param <- new("GOHyperGParams", geneIds = gene_list$ENTREZID,  
            universeGeneIds=univ, annotation="org.Hs.eg.db", ontology="BP")  
  
hyp <- hyperGTest(param)  
  
result<-summary(hyp,categorySize=2000)  
  
write.csv(result,"QD2Sig_GO_BP.csv")
```



FastQC Results



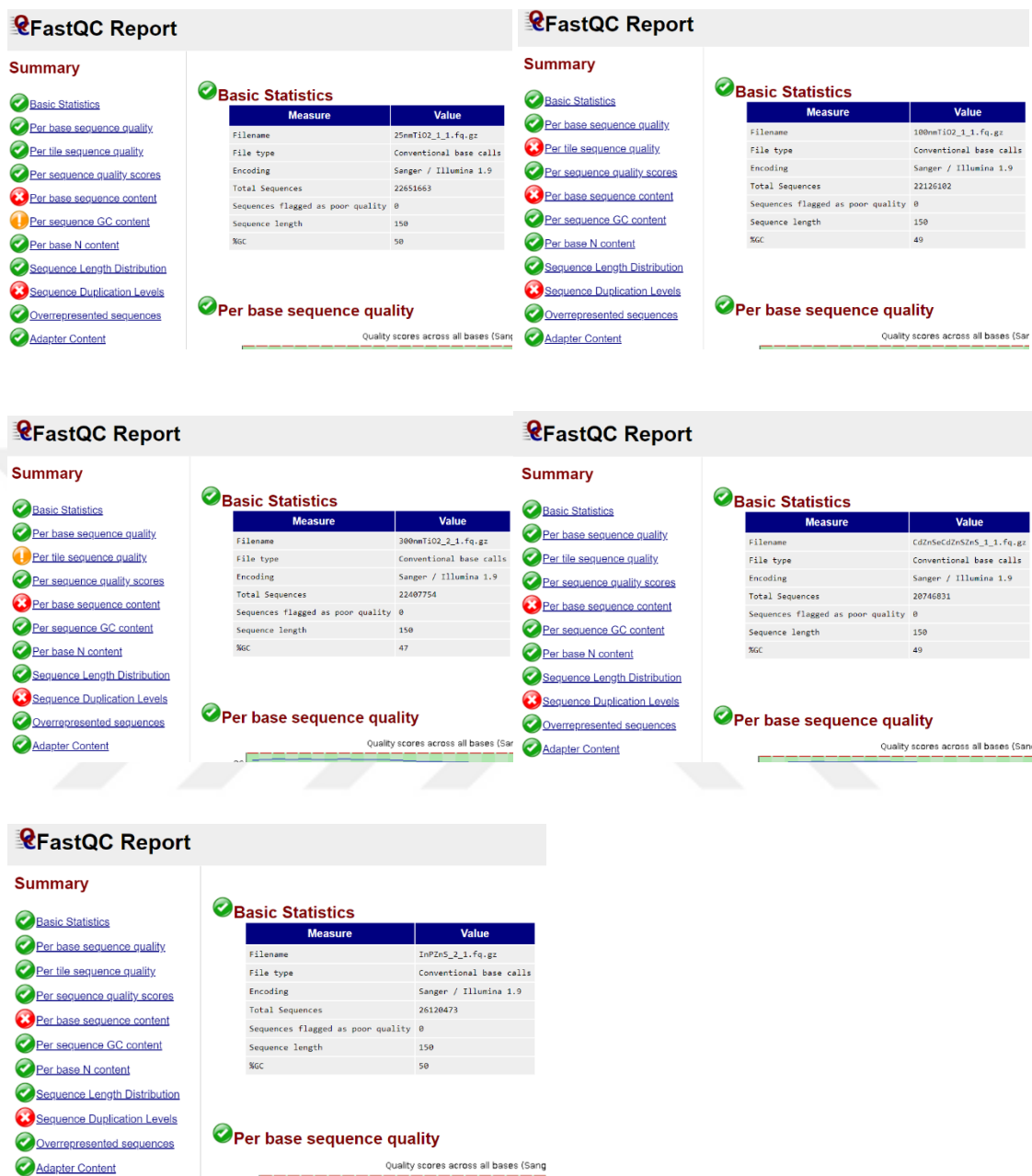


Figure B-1: FastQC quality check results of eleven nanoparticles.

APPENDIX C

List of Top DEGs for each NP

3 nm GOLD			15 nm GOLD			80 nm GOLD		
Gene	pval	log2FC	Gene	pval	log2FC	Gene	pval	log2FC
RMI1	1.89E-06	2.195	TMEM158	3.59E-02	1.671	MT-TG	4.85E-02	1.984
SLF2	9.33E-04	1.790	NREP	2.13E-02	1.605	WFDC21P	3.34E-02	1.926
FHL1	8.03E-03	1.553	NTAN1	4.76E-03	1.465	JDP2	2.36E-02	1.630
AC093673.1	9.09E-04	1.440	FAHD1	3.85E-02	1.259	TLE6	4.49E-02	1.624
ENDOD1	6.40E-03	1.294	PDGFB	4.37E-02	1.203	IGFBP7	3.68E-02	1.597
DMAC2L	1.44E-03	1.223	CUL2	2.70E-02	1.172	AC078850.1	3.45E-03	1.558
NDUFA5	9.80E-03	1.070	AC022400.6	4.82E-03	1.012	TRNP1	3.84E-02	1.505
SUSD2	3.46E-03	0.943	RBM38	3.38E-02	0.941	HS3ST3B1	3.39E-02	1.493
CSTF2	3.59E-03	0.856	SFN	1.01E-02	0.927	KRT8P12	3.30E-02	1.462
ATPAF1	9.61E-03	0.835	AC093673.1	2.73E-02	0.860	OLFML3	4.25E-02	1.316
AC092329.4	2.75E-03	-3.345	TPP2	8.51E-04	-1.434	AQP9	1.20E-02	-2.762
CADM4	1.15E-03	-2.895	DHX35	1.04E-02	-1.114	KNR1	4.40E-02	-2.051
FTH1P11	7.39E-03	-2.354	CPS1	3.96E-02	-0.832	ADH1A	3.13E-02	-1.939
KATNBL1	4.23E-03	-1.739	MT-TW	1.56E-02	-0.806	APOA1	1.02E-02	-1.927
USE1	5.26E-03	-1.673	THUMPDI	4.04E-02	-0.777	ZNF449	3.53E-02	-1.840
NRG1	2.44E-03	-1.512	AC115989.1	3.96E-03	-0.776	TPP2	3.77E-02	-1.546
APOD	7.10E-03	-1.195	SLC23A1	2.35E-02	-0.774	NVL	1.89E-02	-1.208
TNFRSF10A	5.84E-04	-1.053	AL022328.1	1.32E-02	-0.768	VNN1	3.90E-02	-1.206
TPR	5.63E-03	-0.996	ACSM2B	4.77E-02	-0.760	VTN	2.19E-03	-1.145
CYB561	7.70E-03	-0.848	GBF1	1.87E-02	-0.731	CYP4A22	2.26E-02	-1.084

Figure C- 1: List of top 10 upregulated and downregulated genes for gold nanoparticles.

12 nm SILVER			40 nm SILVER			80 nm SILVER		
Gene	pval	log2FC	Gene	pval	log2FC	Gene	pval	log2FC
VRK1	4.86E-02	2.455	MT-TI	1.87E-03	9.844	MT-TK	1.97E-02	6.839
NTAN1	2.48E-02	2.185	MT-TG	1.93E-02	8.789	MT-TG	4.38E-02	6.623
CCN5	3.23E-02	1.818	MT-TK	1.38E-03	8.654	RPS25	4.78E-02	5.341
AC078850.1	3.16E-02	1.704	MT-TF	3.53E-02	8.578	DGAT2	1.54E-02	4.317
CENPH	1.83E-02	1.617	MT-TM	3.14E-02	8.451	MIR34AHG	4.81E-02	4.212
CKLF	7.17E-03	1.526	MT-TL1	2.77E-02	7.461	EZH1P	3.33E-02	3.659
TRIP13	4.68E-02	1.520	MT-TT	1.27E-02	7.452	GHDC	1.44E-02	3.624
DSN1	4.58E-03	1.517	EGR1	2.65E-02	6.966	RARRES2	5.22E-03	3.373
PRRG1	4.73E-02	1.510	RN7SL1	1.41E-03	6.701	CCND1	1.74E-02	3.364
GTPBP3	5.44E-03	1.485	MIR34AHG	6.32E-04	5.772	WFDC21P	2.20E-02	3.276
PID1	4.15E-02	-1.622	CXCL3	4.84E-02	-6.384	AC034213.1	7.73E-03	-5.566
ITIH2	4.50E-02	-1.594	MRPS23	1.20E-02	-6.313	AC009549.1	5.72E-03	-4.748
CXCL2	5.20E-04	-1.575	SDF2L1	4.92E-02	-4.935	CYLD	4.81E-02	-4.425
ABCA7	2.09E-02	-1.521	IL6	1.64E-02	-4.870	AC008760.2	2.72E-02	-4.386
REV1	2.94E-02	-1.318	PSMD9	1.64E-02	-4.846	FAHD2P1	2.57E-02	-3.977
ARHGAP23	3.72E-02	-1.070	TNFAIP6	2.17E-02	-4.608	RASGRP3	1.64E-02	-3.343
TCERG1	3.57E-02	-1.025	MFSD2A	2.78E-02	-4.598	BIN3	1.22E-03	-3.319
C3	1.29E-02	-1.015	KLF4	3.42E-02	-4.285	SNX11	3.52E-02	-3.292
TUT7	4.31E-02	-0.891	S100P	3.49E-03	-4.259	CNOT3	4.18E-02	-3.256
HOMER2	4.96E-02	-0.820	PDIA4	2.80E-02	-4.241	RRAGA	3.44E-02	-3.168

Figure C- 2: List of top 10 upregulated and downregulated genes for silver nanoparticles.

25 nm TITANIUM			100 nm TITANIUM			300 nm TITANIUM		
Gene	pval	log2FC	Gene	pval	log2FC	Gene	pval	log2FC
RRM2	4.19E-02	4.566	TMEM158	2.25E-02	6.074	TMEM158	1.24E-02	3.649
GHDC	3.98E-02	4.051	BEST1	3.57E-02	4.047	IL1RL1	3.73E-02	3.283
TUG1	3.68E-02	3.743	FP236383.5	4.04E-02	4.039	RASD1	3.00E-02	2.864
TPX2	3.00E-02	3.637	AL031663.3	3.69E-02	3.523	MT-TC	4.03E-03	2.840
CCND1	4.25E-02	3.265	SERPINE2	4.51E-02	3.320	PAQR5	3.88E-02	2.731
BCAM	4.13E-03	3.239	CCL2	1.11E-02	3.235	CA9	4.12E-02	2.554
TOP2A	4.11E-02	3.166	PDGFB	7.32E-03	3.159	SLCO4A1	1.17E-02	2.372
AL391427.1	3.91E-02	3.093	EDN2	1.00E-02	2.908	CDCP1	3.02E-02	2.249
FCSK	2.03E-02	3.039	MASP1	9.75E-03	2.495	HS3ST3B1	4.95E-02	2.213
IGFBP2	4.36E-02	3.009	SPIDR	3.76E-02	2.375	CD24P4	2.19E-02	2.208
HERPUD1	2.69E-02	-5.830	ITIH2	1.66E-02	-2.778	TTR	4.04E-02	-5.367
DDIT3	4.22E-02	-4.874	AGPAT1	2.42E-03	-2.740	RPS26P6	7.63E-03	-3.205
AC012181.2	4.56E-02	-4.865	NR1H4	1.90E-02	-2.198	BEST1	1.99E-02	-3.038
SELENOS	1.57E-03	-4.667	PCMTD2	2.81E-02	-2.013	AL355032.1	2.54E-02	-2.837
TRIB3	3.26E-02	-4.558	GSDMD	1.10E-02	-1.985	AC078817.1	4.35E-02	-2.782
MANF	2.85E-02	-4.513	MAN1A2	1.42E-02	-1.894	KNG1	3.03E-02	-2.733
NDUFA4L2	4.33E-02	-4.400	RFFL	2.99E-02	-1.843	FICD	3.22E-02	-2.680
SELENOK	3.22E-02	-4.133	ZBTB45	2.58E-02	-1.785	GJB1	4.97E-02	-2.656
KCTD11	4.73E-02	-4.068	RELCH	1.25E-02	-1.757	TF	3.58E-02	-2.548
SERP1	3.55E-02	-3.886	GABPB1-AS1	3.73E-02	-1.687	SDSL	2.53E-02	-2.469

Figure C- 3: List of top 10 upregulated and downregulated genes for titanium nanoparticles.

Quantum Dot 1			Quantum Dot 2		
Gene	pval	log2FC	Gene	pval	log2FC
RN7SL3	2.93E-02	6.807	MT-TI	2.42E-02	6.330
RN7SL4P	2.84E-02	4.818	MT-TL1	3.14E-02	6.305
KRT19	1.31E-03	3.176	MT-TV	3.26E-02	5.717
GALNT15	5.42E-03	2.991	MT-TM	4.17E-02	5.660
HSPA1A	1.62E-02	2.617	MT-TF	4.26E-02	4.730
NEBL	3.38E-02	2.532	MT-TT	1.60E-02	4.192
ABAT	3.30E-02	2.510	TMEM176A	3.43E-02	4.188
FCSK	4.37E-02	2.434	DPT	1.25E-02	3.899
CITED4	1.92E-02	2.416	MT-TG	4.11E-02	3.853
RNF213	4.10E-02	2.399	TBC1D4	1.89E-02	3.837
OLA1	4.55E-02	-3.130	BEST1	2.93E-02	-6.682
APOD	4.59E-02	-2.697	AC006064.4	1.09E-02	-4.759
AC073861.1	2.74E-03	-2.625	SDF2L1	3.08E-02	-4.366
TP53BP1	2.35E-02	-2.616	CXCL3	3.01E-03	-4.164
SEC11C	3.39E-02	-2.579	F11R	2.14E-02	-4.025
LSR	4.98E-04	-2.432	AC012181.2	2.32E-02	-3.969
URM1	3.64E-02	-2.385	DDIT3	2.52E-02	-3.676
AL121594.1	4.41E-02	-2.315	TFIP11	2.49E-02	-3.660
U2AF1	2.43E-02	-2.173	PPP1R15A	2.50E-02	-3.424
GBP3	6.91E-03	-2.160	SELENOS	4.77E-02	-3.373

Figure C- 4: List of top 10 upregulated and downregulated genes for quantum dots.