Using Machine Learning Algorithms to Predict the likelihood of Polycystic Ovarian Syndrome based on Demographic, Clinical and Lifestyle Factors

CUNY SPS Data 698 - Analytics Masters Research Project

Leticia Salazar

January 8, 2024

Contents

YouTube Presentation

Abstract:

Polycystic Ovarian Syndrome (PCOS) stands as a prevalent endocrine disorder affecting women of reproductive age worldwide, presenting a confluence of hormonal imbalances, reproductive irregularities, and potential metabolic complications. Characterized by irregular menstrual cycles, hyperandrogenism, and polycystic ovaries, PCOS poses multifaceted challenges that extend beyond reproductive health, encompassing metabolic disturbances and psychological implications. This multifaceted syndrome, influenced by intricate interactions among genetic predispositions, hormonal imbalances, and environmental elements, presents with diverse manifestations among those affected. The clinical landscape of PCOS often requires a comprehensive, multidisciplinary approach, incorporating lifestyle modifications, pharmacological interventions, and personalized treatments to address symptoms and reduce associated health risks. Despite ongoing research efforts, unraveling the exact cause and identifying the most effective management approaches for PCOS remains an evolving field of investigation within modern medicine. This research seeks to explore current studies related to PCOS diagnosis by acquiring datasets from Kaggle.com and assessing the efficiency of diverse machine learning algorithms, encompassing Decision Tree (rpart), Random Forest (randomForest), Gradient Boosting Machines (xgboost or gbm), Support Vector Machines (SVM; e1071), Neural Networks (neuralnet), and K-Nearest Neighbors (knn or class). Upon data exploration and preprocessing, models were constructed using the six distinct algorithms. The analysis revealed that Decision Tree 2, SVM 2, SVM 1, and Random Forest 2 exhibited relatively superior performance in discerning between PCOS and non-PCOS cases within this dataset. On the other hand, k-Nearest Neighbor 1, Random Forest 1, Gradient Boost Machines 2 and both Neural Networks models displayed lower predictive accuracies.

Key words:

Polycystic Ovarian Syndrome, PCOS, Women's Health, Machine Learning Algorithms, Predictive Modeling

The Problem:

Polycystic Ovarian Syndrome (PCOS) is a hormonal imbalance disorder affecting women of reproductive age. Determining the precise global count of women affected by PCOS poses challenges due to many cases remaining undiagnosed. However, the World Health Organization (WHO) estimates that approximately 3.4% of women are affected [WHO, 2023]. While this percentage might seem relatively small, considering that women constitute 49.7% of today's population [Knoema, 2022], nearly 13% of them fall within the reproductive age bracket [MarchofDimes, 2022]. This data suggests that approximately 17.5 million women report suffering from PCOS. Beyond the physical and emotional toll PCOS exacts, it also disrupts ovarian function, leading to challenges in maintaining a healthy menstrual cycle and can result in the formation of cysts, ultimately impacting fertility. It's important to note that the prevalence of PCOS varies by region and ethnic groups, with some studies suggesting higher rates of PCOS in certain populations [Engmann, 2017]. Early identification of risk factors associated with PCOS can assist in timely interventions and lifestyle adjustments. Tailoring suggestions or treatments based on individualized risk profiles has the potential to enhance patient outcomes. Employing predictive models can play a pivotal role in increasing awareness regarding PCOS risk factors and preventive measures. However, limitations may exist in accessing comprehensive and varied datasets containing accurate demographic, clinical, and lifestyle data [Thakre, 2020]. Addressing these challenges involves the development of a robust predictive models using machine learning techniques. Such efforts holds the promise of significantly contributing to the identification of individuals at risk of PCOS, thereby enabling early interventions and guiding personalized healthcare strategies for improved management of the condition. This project will investigate publicly available datasets that will be used to develop machine learning models that predicts the probability or risk of an individual having or developing PCOS based on demographic information (age, ethnicity, geographical location), clinical data (hormonal levels, BMI, menstrual irregularities), and lifestyle factors (dietary habits, exercise routine, stress levels).

Objectives:

Construct predictive models using machine learning techniques that utilize a dataset comprising demographic, clinical, and lifestyle variables as features and PCOS diagnosis as the target variable. Machine learning algorithms have shown promise in advancing our understanding of the disease and improving its diagnosis and treatment.

I anticipate answering the following questions with my data:

1. Are there commonalities women with and without PCOS have that can be easily dismissed as normal?
2. Are there differences for women of different race/ethnic background when it comes to having PCOS? What about women without PCOS?
3. What is the likelihood of a woman developing PCOS based on her age, ethnicity, and BMI history?
4. Can we predict the risk of insulin resistance, diabetes, and cardiovascular disease in women with PCOS based on their medical history, hormone levels, and lifestyle factors?
5. Can we predict the likelihood of successful pregnancy outcomes in women with PCOS based on their age, weight, hormone levels, and treatment history?
6. Can we predict the long-term health outcomes and quality of life of women with PCOS based on their age, lifestyle factors, hormone levels, and treatment history?

Literature Review:

For this project, my emphasis was on discovering literature reviews that validate the relevance of the dataset utilized alongside the chosen machine learning algorithms. In pursuit of this goal, I've come across various articles that make reference to the dataset [Kottarathil, 2020]. These articles investigate studies associated with PCOS, presenting findings aimed at providing a deeper comprehension of the dataset I'm analyzing.

The literature reviews collectively delve into various aspects of Polycystic Ovarian Syndrome (PCOS), employing diverse methodologies and approaches for diagnosis, classification, understanding clinical manifestations, and proposing potential treatments. Researchers across these studies have primarily utilized data mining, machine learning, and clinical investigations to address the complexity of PCOS. Below are the key characteristics, achievements, advantages, and drawbacks across these reviews:

Common Focus Areas: Multiple studies emphasize the use of machine learning algorithms, such as Naïve Bayes, Decision Trees, Artificial Neural Networks, Support Vector Machines, and ensemble methods, for PCOS diagnosis. They explore the accuracy and predictive power of these models using diverse datasets, including clinical, lifestyle, and physiological parameters. Investigations into clinical parameters and anthropometric measures aim to identify potential predictors or indicators of PCOS, such as hormonal imbalances, insulin resistance, metabolic traits, obesity, and associated risks like infertility and cardiovascular issues. Studies examine the diverse clinical presentations and phenotypic variations within PCOS, shedding light on how different subgroups may manifest the syndrome and respond to various treatments.

Achievements: Machine learning models, particularly those utilizing Convolutional Neural Networks (CNNs) and ensemble methods, have shown high accuracy in diagnosing PCOS. The findings revealed that CNN models performed best, achieving accuracies ranging from 85% to 98.12% in different studies [Anda, 2022]. These models effectively utilize diverse datasets, ranging from ultrasound images to clinical parameters. Research has identified several potential predictive factors for PCOS, including hormonal markers, lifestyle attributes, and metabolic indicators, offering insights into early detection and tailored treatment strategies. The study done by Aggarwal, S., & Pandey, K. (2023) used supervised learning algorithms (like

random forest, gradient boosting) to assess performance metrics, indicating that these crucial features offer high accuracy in identifying PCOS. Unsupervised learning (K-means clustering) corroborates these findings, suggesting that these key features are pivotal for PCOS analysis. In another study, Tiwari (2022) uses machine learning to screen PCOS patients based on non-invasive parameters. It employed various algorithms like SVM, Decision Trees, Random Forest, etc., for classification, finding that Random Forest achieved the highest accuracy of 93.25%. Overall,studies evaluating PCOS awareness among women have highlighted the importance of education and awareness campaigns in enhancing understanding and facilitating early diagnosis.

Advantages: Machine learning algorithms offer promising accuracy rates, particularly CNNs and ensemble models, in diagnosing PCOS using various non-invasive parameters. Understanding phenotypic variations aids in tailoring treatments based on specific subgroups, potentially improving patient outcomes and management. Efforts toward identifying early markers or predictive factors can facilitate early intervention and lifestyle modifications, mitigating long-term health risks associated with PCOS.

Drawbacks and Recommendations: Some studies may suffer from limited sample sizes or datasets, impacting the generalizability of findings. Larger and more diverse datasets are recommended for robust model development and validation [Goodarzi, 2015]. While certain machine learning models showcase high accuracy, the variability in dataset characteristics and preprocessing methods could influence their effectiveness across different populations or settings [Tiwari, 2022]. The multifaceted nature of PCOS, influenced by both genetic and environmental factors, presents challenges in pinpointing a singular cause or standard diagnostic criteria.

The collective body of literature reviews signifies advancements in PCOS diagnosis, understanding clinical manifestations, and potential avenues for tailored treatments. The use of machine learning models, especially those employing CNNs and ensemble methods, showcases significant promise in accurate PCOS diagnosis based on non-invasive parameters. However, further research is necessary, emphasizing larger and more diverse datasets, refined models, and a multidisciplinary approach to fully comprehend the complexity of PCOS and enhance diagnostic and treatment strategies.

Dataset:

The Polycystic ovary syndrome (PCOS) dataset, available on Kaggle.com, is comprised of two csv files labeled `PCOS_data_without_infertility` and `PCOS_infertility`. In total, these files encompass 48 variables and 541 data entries all collected from 10 different hospitals across Kerala, India. The dataset contains all physical and clinical parameters to determine PCOS and infertility related issues.

Full description of the variables below:

- Units used range from imperial to metric system of measurement
- For Yes | No questions
  - Yes = 1
  - No = 0

Variables	Description
"Sl..No"	unique identification number assigned to each entry
"Patient.File.No."	file number for each patient's record.
"PCOS..Y.N."	presence or absence of PCOS
"I...beta.HCG.mIU.mL."	pregnancy hormone case I measured in milli-international units per liter (mIU/L)

Variables	Description
"II....beta.HCG.mIU.mL."	pregnancy hormone case II measured in milli-international units per liter (mIU/L)
"AMH.ng.mL."	detects ovarian reserve (egg count)
"Age..yrs."	age of patient in years
"Weight..Kg."	weight of patient in kg
"Height.Cm."	height of patient in cm
"BMI"	body mass index
"Blood.Group"	Blood Groups: A+ = 11, A- = 12, B+ = 13, B- = 14, O+ = 15, O- = 16, AB+ = 17, AB- = 18
"Pulse.rate.bpm."	beats per minute
"RR..breaths.min."	respiration rates per minute
"Hb.g.dl."	hemoglobin concentration measured in grams per deciliter (g/dL).
"Cycle.R.I."	cycle Regularity Index used to assess the regularity or irregularity of menstrual cycles in women: 4 indicates irregular menstrual cycle, 2 indicates a regular menstrual cycle
"Cycle.length.days."	length of menstrual cycle
"Marraige.Status..Yrs."	years married
"Pregnant.Y.N."	pregnant yes or no
"No..of.aborptions"	number of abortions
"FSH.mIU.mL."	follicle stimulating hormone measured in milli-international units per liter (mIU/L)
"LH.mIU.mL."	luteinizing hormone (increases during ovulation) measured in milli-international units per liter (mIU/L)
"FSH.LH"	ratio between Follicle-Stimulating Hormone (FSH) and Luteinizing Hormone (LH)
"Hip.inch."	measurement of hips in inches
"Waist.inch."	measurement of waist in inches
"Waist.Hip.Ratio"	ratio of measurement of waist and hip
"TSH..mIU.L."	thyroid stimulating hormone measured in milli-international units per liter (mIU/L)
"AMH.ng.mL."	Anti-Müllerian Hormone (AMH) measured in nanograms per milliliter (ng/mL); a marker used in reproductive medicine to assess ovarian reserve
"PRL.ng.mL."	Prolactin measured in nanograms per milliliter (ng/mL); a hormone produced by the pituitary gland
"Vit.D3..ng.mL."	Vitamin D3 measured in nanograms per milliliter (ng/mL); is essential for bone health, immune function, and various other bodily processes.
"PRG.ng.mL."	Progesterone measured in nanograms per milliliter (ng/mL); a hormone involved in the menstrual cycle, pregnancy, and maintaining the uterine lining for a developing embryo.
"RBS.mg.dl."	Random Blood Sugar measured in milligrams per deciliter (mg/dL); it represents the level of glucose (sugar) present in the blood at a random time, without fasting.
"Weight.gain.Y.N."	weight gain yes or no
"hair.growth.Y.N."	hair growth yes or no (hirsutism)
"Skin.darkening..Y.N."	darkening of skin yes or no
"Hair.loss.Y.N."	hair loss yes or no
"Pimples.Y.N."	pimples (acne) yes or no

Variables	Description
"Fast.food..Y.N."	consumption of fast food yes or no
"Reg.Exercise.Y.N."	regularly exercise yes or no
"BP._Systolic..mmHg."	systolic blood pressure measured in millimeters of mercury (mmHg)
"BP._Diastolic..mmHg."	diastolic blood pressure measured in millimeters of mercury (mmHg)
"Follicle.No...L."	number of follicles on left ovary
"Follicle.No...R."	number of follicles in right ovary
"Avg..F.size..L...mm."	average size of follicles in left ovary measured in millimeters (mm)
"Avg..F.size..R...mm."	average size of follicles in right ovary measured in millimeters (mm)
"Endometrium..mm."	size of the endometrial thickness in millimeters (mm)

Methodology:

Upon importing the data into R, I conducted an assessment to gain a comprehensive understanding of the dataset. The exploration revealed a necessity for substantial data preparation before commencing model construction. Utilizing the `skimr` library, I generated concise summaries for both datasets, which indicated the presence of character and numeric column types without any missing values. Employing the `Data Explorer` library, histograms were created to examine the distribution of variables in both datasets. However, these distributions did not display any discernible patterns or distinct shapes.

The data preparation phase involved standardizing the `pcos` and `pcos2` datasets as numeric due to variations in variable classes. While confirming the absence of missing data, it was identified that `pcos2` contained a few variables with missing values. Upon obtaining a comprehensive overview of both datasets, the procedure involved eliminating unnecessary columns, renaming columns for enhanced readability, and merging the datasets. Additionally, transformations were applied, converting `Height` from centimeters to meters, `Hip` and `Waist` from inches to centimeters. Subsequently, missing values in columns such as `BMI`, `Waist_Hip_Ratio`, and `FSH_LH` were computed and replaced. For `Married_yrs`, `AMH_ngmL`, and `Fast_food`, the missing values were substituted with the median values, as these replacements did not significantly impact the data distribution. Detecting outliers was crucial and accomplished through boxplots for visual representation. Despite their presence, I chose to retain these outliers, acknowledging their importance in representing natural variations within the population. Additional visualizations were created to visualize trends within the refined `pcos_cleaned` dataset.

Building the models involved the development of six distinct machine learning algorithms: Decision Trees, Random Forest, Gradient Boost Machines, Support Vector Machines, Neural Networks, and k-Nearest Neighbor. I initiate by dividing the dataset into training and validation subsets tailored for machine learning models. The training subset is utilized for model training, while the validation subset assesses its performance. Employing a 75:25 ratio strikes a balance, ensuring adequate data for effective model training and a sizable validation set for robust evaluation.

For each algorithm, the following procedures were executed:

1. Establish a cross-validation configuration, employing the `PCOS` variable as the target against the entire dataset. The split data designated `train` and `valid` labeled the training and testing subsets, respectively. Additionally, another cross-validation setup centered on the `PCOS` target variable, utilizing pivotal contributing variables: `Follicle_NoL`, `Follicle_NoR`, `Hair_growth`, `Skin_darkening`, and `Weight_gain`.

2. After the model creation using the `train` dataset, predictions were generated for the model utilizing the `valid` data. The output encompassed a confusion matrix and associated statistics affirming the predictive output for the target variable (`PCOS`). This also included metrics providing insights into the model's efficacy in classifying positive and negative cases accurately in binary classification.

3. Evaluate the individual contributions of each variable within the model and, if necessary, visualize their significance.

4. Retrieve the accuracy metric for subsequent comparison with other models.

The procedure was replicated for all six algorithms on two occasions: first, using the complete dataset, and second, utilizing the reduced dataset containing the highest contributing variables. The culmination of the project involved aggregating all accuracies obtained from the models to compare the performance of each model in differentiating between PCOS and non-PCOS cases across the datasets.

Assumptions:

While there's limited information available in the medical field and even less data sets available to analyze, I have some concerns on being successful in predicting a PCOS diagnosis. Yet there are justifications exploring PCOS in depth:

- Identify diagnostic biomarkers that can distinguish PCOS patients from healthy individuals or those with other disorders. These biomarkers can aid in earlier diagnosis and better management of the disease.
- Predict the likelihood of disease progression and the risk of developing complications, such as diabetes and cardiovascular disease, in PCOS patients. This information can guide treatment decisions and improve patient outcomes.
- Develop personalized treatment plans for PCOS patients based on their individual characteristics and medical history. This approach can lead to more effective and targeted interventions.
- Integrate data from various sources, such as electronic health records, imaging studies, and genetic analyses, to provide a more comprehensive understanding of PCOS. This can help identify new pathways involved in the disease and potential targets for therapy.
- Aid in the design and analysis of clinical trials, leading to more efficient and informative studies. This can accelerate the development of new treatments for PCOS.

Early diagnosis and management of PCOS can lead to better health outcomes, improved quality of life, and reduced long-term health risks. Therefore, predicting PCOS diagnosis can have several societal benefits, including:

- Predicting PCOS diagnosis can help healthcare providers identify women at risk of developing PCOS and intervene early with appropriate treatment, such as lifestyle modifications and medication, to prevent or minimize the long-term health consequences of the disorder.
- Early diagnosis and treatment of PCOS can help manage symptoms such as irregular periods, infertility, acne, and excess hair growth, leading to improved physical and mental health outcomes for affected women.
- By predicting PCOS diagnosis and intervening early, healthcare providers can prevent or reduce the need for more expensive treatments or surgeries later in life, resulting in cost savings for individuals, healthcare systems, and society.
- Predicting PCOS diagnosis can increase awareness of the disorder among healthcare providers, patients, and the public, leading to more education, research, and advocacy efforts aimed at improving PCOS diagnosis, treatment, and management.
- Early intervention and management of PCOS can improve the quality of life for affected women, leading to increased productivity, better mental health, and greater overall well-being.

Overall, I'll be able to explore the insights into PCOS pathophysiology, diagnosis, and treatment. Their use in PCOS research can lead to more personalized and effective care for patients with this complex disorder.

Experimentation and Results:

Data Exploration and Preparation Once I started the data exploration process I uncovered the following:

- In total, the dataset is composed of 48 variables and 541 data entries
- The data was collected from 10 different hospitals across Kerala, India
- There seemed to be no missing variables
- The initial histograms of `pcos` and `pcos2` did not indicate a distinct pattern or discernible shape.

The data preparation process included renaming columns and addressing missing values. This included handling missing values by calculating the median as a replacement strategy for columns like `Marraied_yrs`, `AMH_ngmL` and `Fast_food`. Additionally, for columns such as `BMI`, `FSH_LH`, and `Waist_Hip_Ratio`, the values were computed. Specific columns were converted to the metric system, and redundant or unnecessary columns were removed. These steps aim to streamline data management before merging the two datasets.

The final results are seen in `pcos_cleaned` below:

Table 4. `pcos_cleaned` dataset

PCOS	Age_yrs	Weight	Height	BMI	Blood_Group	Pulse_rate_bpm	RR_breaths_min	Hb_gdl	Cycle_RI	Cycle_length_days	Married_yrs	Pregnant
0	28	44.6	1.5	19.8	15	78	22	10.5	2	5	7.0	0
0	36	65.0	1.6	25.4	15	74	20	11.7	2	5	11.0	1
1	33	68.8	1.7	23.8	11	72	18	11.8	2	5	10.0	1
0	37	65.0	1.5	28.9	13	72	20	12.0	2	5	4.0	0
0	25	52.0	1.6	20.3	11	72	18	10.0	2	5	1.0	1
0	36	74.1	1.7	25.6	15	78	28	11.2	2	5	8.0	1
0	34	64.0	1.6	25.0	11	72	18	10.9	2	5	2.0	0
0	33	58.5	1.6	22.9	13	72	20	11.0	2	5	13.0	1
0	32	40.0	1.6	15.6	11	72	18	11.8	2	5	8.0	0
0	36	52.0	1.5	23.1	15	80	20	10.0	4	2	4.0	0
0	20	71.0	1.6	27.7	15	80	20	10.0	2	5	4.0	1

The final results are seen in `pcos_cleaned` below:

Pregnant	No_of_abortions	IbetaHCG_mIUmL	IIbetaHCG_mIUmL	FSH_mIUmL	LH_mIUmL	FSH_LH	Hip	Waist	Waist_Hip_Ratio	TSH_mIUmL	AI
0	0	2.0	1.990	8.0	3.7	2.2	91.4	76.2	0.83	0.7	
1	0	60.8	1.990	6.7	1.1	6.2	96.5	81.3	0.84	3.2	
1	0	494.1	494.080	5.5	0.9	6.3	101.6	91.4	0.90	2.5	
0	0	2.0	1.990	8.1	2.4	3.4	106.7	91.4	0.86	16.4	
1	0	801.5	801.450	4.0	0.9	4.4	94.0	76.2	0.81	3.6	
1	0	238.0	1.990	3.2	1.1	3.0	111.8	96.5	0.86	1.6	
0	0	2.0	1.990	2.9	0.3	9.2	99.1	83.8	0.85	1.5	
1	2	100.5	100.510	4.9	3.1	1.6	111.8	96.5	0.86	12.2	
0	1	2.0	1.990	3.8	3.0	1.2	99.1	88.9	0.90	1.5	
0	0	2.0	1.990	2.8	1.5	1.9	101.6	96.5	0.95	6.7	
1	2	158.5	158.510	4.9	2.0	2.4	99.1	88.9	0.90	1.6	

The final results are seen in `pcos_cleaned` below:

AMH_ngmL	PRL_ngmL	Vit_D3_ngmL	PRG_ngmL	RBS_mgdl	Weight_gain	Hair_growth	Skin_darkening	Hair_loss	Pimples	Fast_food	Reg_Exerci
2.1	45.2	17.1	0.6	92.0	0	0	0	0	0	1	
1.5	20.1	61.3	1.0	92.0	0	0	0	0	0	0	
6.6	10.5	49.7	0.4	84.0	0	0	0	1	1	1	
1.2	36.9	33.4	0.4	76.0	0	0	0	0	0	0	
2.3	30.1	43.8	0.4	84.0	0	0	0	1	0	0	
6.7	16.2	52.4	0.3	76.0	1	0	0	1	0	0	
3.0	26.4	42.7	0.5	93.0	0	0	0	0	0	0	
1.5	4.0	38.0	0.3	91.0	1	0	0	0	0	0	
1.0	19.0	21.8	0.3	116.0	0	0	0	0	0	0	
1.6	11.7	27.7	0.2	125.0	0	0	0	0	0	0	
4.5	13.5	18.1	0.4	108.0	0	0	0	0	0	0	

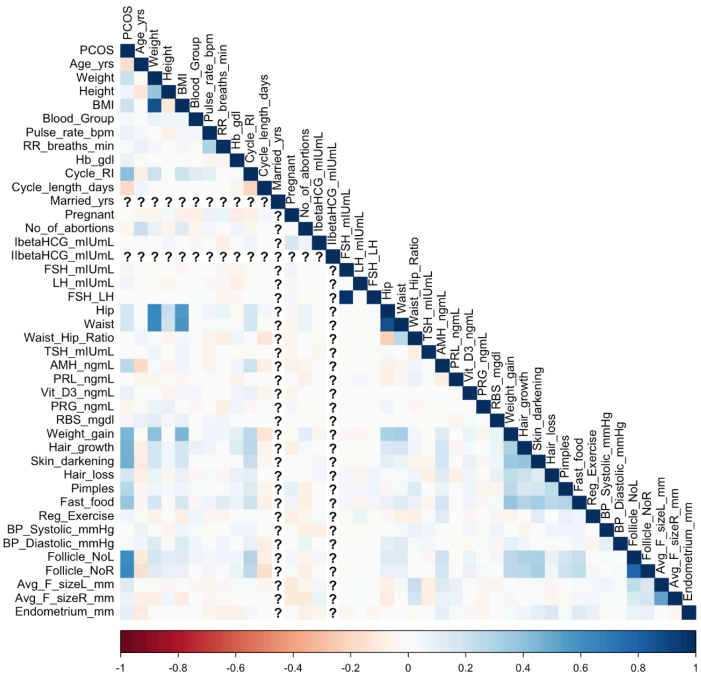
The final results are seen in `pcos_cleaned` below:

Fast_food	Reg_Exercise	BP_Systolic_mmHg	BP_Diastolic_mmHg	Follicle_NoL	Follicle_NoR	Avg_F_sizeL_mm	Avg_F_sizeR_mm	Endometrium_mm
1	0	110	80	3	3	18.0	18.0	8.5
0	0	120	70	3	5	15.0	14.0	3.7
1	0	120	80	13	15	18.0	20.0	10.0
0	0	120	70	2	2	15.0	14.0	7.5
0	0	120	80	3	4	16.0	14.0	7.0
0	0	110	70	9	6	16.0	20.0	8.0
0	0	120	80	6	6	15.0	16.0	6.8
0	0	120	80	7	6	15.0	18.0	7.1
0	0	120	80	5	7	17.0	17.0	4.2
0	0	110	80	1	1	14.0	17.0	2.5
0	0	110	80	7	15	17.0	20.0	6.0

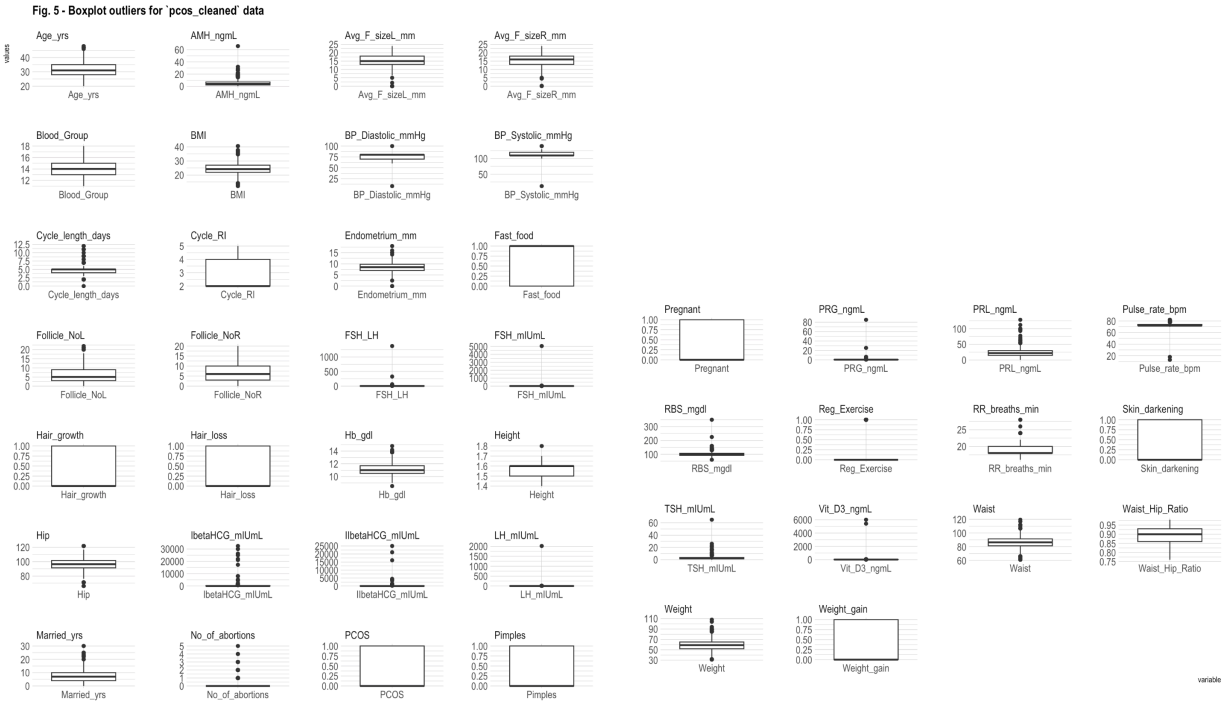
Once the data was cleaned named `pcos_cleaned`, another `DataExplorer` histogram was created; each variable demonstrated variations, displaying either a normal distribution, right-skewed, left-skewed, or no discernible pattern. A correlation plot was then created to measure the degree of linear relationship within the dataset. Based on my findings, there are not many variables that exhibit a notably strong positive or negative correlation, although some variables do fall within these categories to some extent.

- Positive correlations between: `BMI` and `Weight`, `Weight_gain`, `Hair_growth`, `Skin_darkening` and `PCOS` and `Follicle_NoL`, `Follicle_NoR`, and `PCOS` to name a few.
- Negative correlations between:`Waist_Hip_Ratio` and `Hip`, `AMH_ngmL` and `Age_yrs`, and `Follicle_NoR`, `Follicle_NoL` and `Age_yrs`.

Fig. 4 Correlation plot of `pcos_cleaned` data

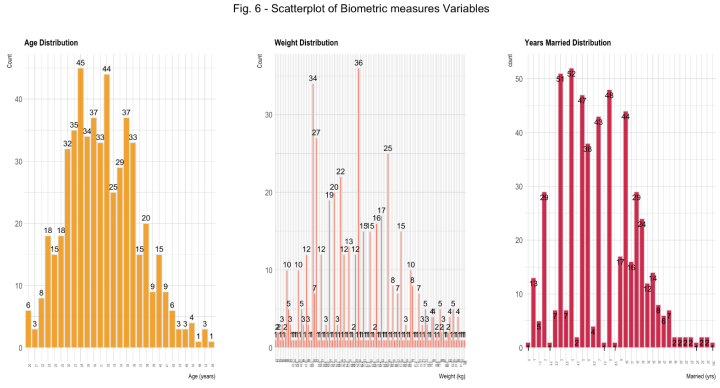


Detecting and managing outliers is essential for maintaining the integrity and reliability of data-driven analyses. With the help of boxplots we can identify outliers across a complete dataset. According to the visualizations provided, a handful of outliers are evident. Variables that included outliers were mainly hormonal markers such as: `AMH_ngml`, `Endometrium_mm`, `FSH_LH`, `FSH_mIUmL`, `Hb_gdl`, `IbetaHCG_mIUmL`, `IIbetaHCG_mIUmL`, `LH_mIUmL`, `PRL_ngml`, and `TSH_mgdl`. Considering the dataset's relatively small size, I've chosen to retain these outliers to ensure the inclusion of every individual data point as it represents natural variations in the population.



As the datasets initially contained missing information during the data exploration phase, additional visualizations were generated specifically for the `pcos_cleaned` dataset. These visualizations include a series of histograms, barcharts and scatterplots.

- There is a wide range of distribution between the age of these women most falling between 23 to 38 years old. Similarly, weight also has a wide range of distribution.
- Most of the women in this study are between 1.5 - 1.6 meters tall (4'9" - 5'2").
- The distribution of years married also varies where the majority of the women fall between the first 10 years.

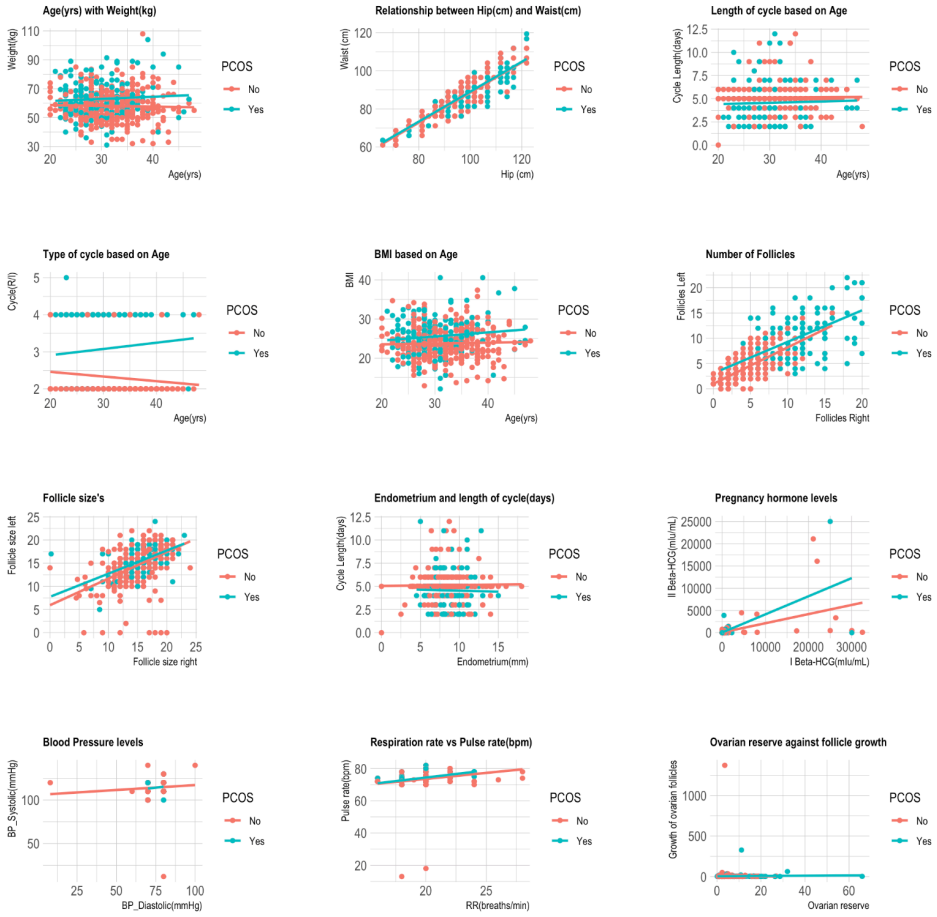


Various physiological and hormonal patterns within the dataset and related to PCOS show that:

- The age and weight of the women are distributed evenly throughout as opposed to being clustered in a certain age and weight group.

- Hip to waist counts increase for those with and without PCOS.
- Cycle length seems to be very consistent regardless of PCOS diagnosis.
- Women with PCOS experience more irregular periods with a few with no PCOS do as well.
- With or without PCOS women's BMI fluctuates in women of all ages.
- The distribution of number of follicles is greater for those women without PCOS.
- The distribution of size in follicles left or right don't differ as greatly as I thought for women with and without PCOS.
- The variance in endometrial thickness between women with PCOS (thinner) and those without PCOS (thicker) evolves as the menstrual cycle progresses. This discrepancy arises due to hormonal imbalances. In women affected by PCOS, the absence of regular menstrual cycles leads to an unaltered endometrial lining, contrasting with the changes observed in women without the condition.
- Pregnancy hormones are clustered with the exceptions of some outliers which were initially noted.
- Blood pressure levels, Pulse rate and Respiration rate for women with and without PCOS are in normal range.
- The quantity of eggs and the development of follicles show similar patterns between women self-reporting PCOS and those without, yet there is no discernible correlation between them.

Fig. 7 - Scatterplot of variables with PCOS (Y/N) as factor



Out of 541 women:

- 32.72% reported to have PCOS
- 38.08% of women reported to being pregnant
- 37.71% reported to experience weight gain
- 27.36% reported to experience hair growth
- 30.68% experience skin darkening
- 45.29% reported to experience hair loss
- 48.98% reported to experience pimples
- 51.57% reported to have fast food
- 24.77% reported to exercise regularly

Taking into account the presence of PCOS within the study:

- Women possessing blood types A+, B+, and O+ reported a higher incidence of PCOS compared to those with blood types A-, B-, O-, AB+, and AB-.
- Being that PCOS and infertility are linked, there are 11.83% of women who reported to being pregnant with PCOS.
- 22.37% of women reported to experience weight gain with PCOS.
- 18.67% of women reported to experience hair growth with PCOS.
- 20.33% of women reported to experience skin darkening with PCOS.
- 18.85% of women reported to experience hair loss with PCOS.
- 22.74% of women reported to experience pimples with PCOS.
- 25.13% of women reported to consume fast food with PCOS.
- 9.43% of women reported to regularly exercise with PCOS.
- Not surprised to see how fluctuated the cycle length is especially for women without PCOS since they tend to have a more regular cycle.
- Abortions were also included in this study but there's no evidence that it's influenced by PCOS; more women without PCOS experience 1 or more abortions.

Within hormonal markers:

- Vitamin D3 levels are spread out for those who reported with or without PCOS.
- FSH/LH levels are right skewed for those who reported with or without PCOS.
- TSH levels are also spread out for women who reported to have PCOS and those who don't.
- Hemoglobin levels seem to be higher for those who reported to have PCOS.
- PRL levels (prolactin in the blood) are consistent at 1 or 2 for those with or without PCOS.
- RBS (random glucose) is fairly distributed with those with and without PCOS.

Fig. 8 - Bar charts of Bloodwork variables

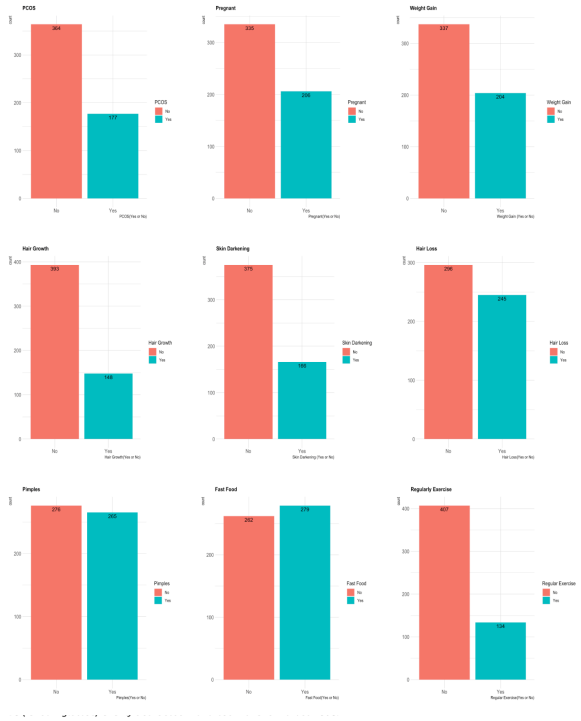
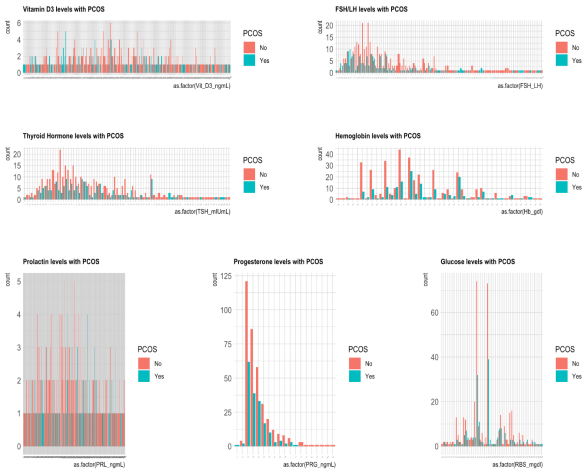


Fig. 9 - Bar charts of yes or no variables

Fig. 10 - Bar charts of Biometrics Measures including PCOS (Y/N) as factor



Model Building: The model building involved the creation of 6 algorithms: Decision Tree (rpart), Random Forest (randomForest), Gradient Boosting Machines (xgboost or gbm), Support Vector Machines (e1071), Neural Networks (neuralnet), and K-Nearest Neighbors (kknn or class). For each algorithm, two models were generated. The first model was developed using the entire dataset with the target variable `PCOS`, while the second models were established using only the most influential variables, `Follicle_NoL`, `Follicle_NoR`, `Hair_growth`, `Skin_darkening`, `Weight_gain` and the target variable `PCOS`.

Decision Tree:

- Model 1 `Decision Tree 1` was run against the whole dataset and yield an accuracy of 85.93%
- Model 2 `Decision Tree 2` was run against the 6 most influential variables and yield and accuracy of 91.85%

Fig. 11 - Decision Tree with entire dataset

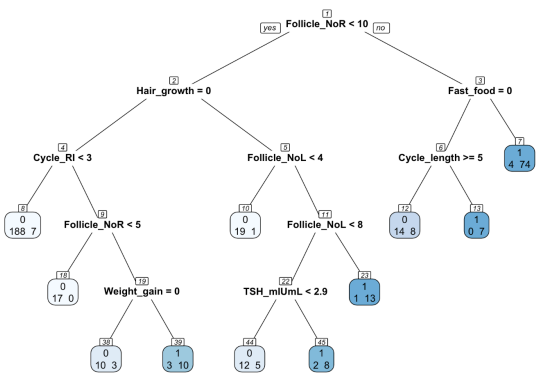
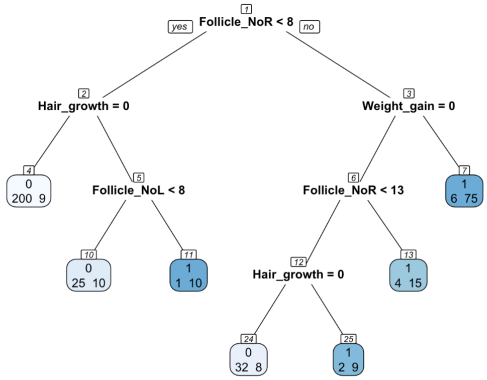


Fig. 12 - Second Decision Tree with 6 variables



Random Forest:

- Model 3 `Random Forest 1` was run against the whole dataset and yield an accuracy of 62.96%
- Model 4 `Random Forest 2` was run against the 6 most influential variables and yield and accuracy of 90.37%

Fig. 13 - Feature importance of first random forest model

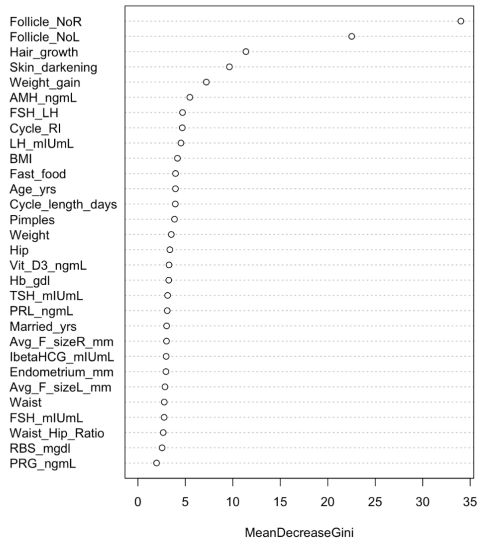
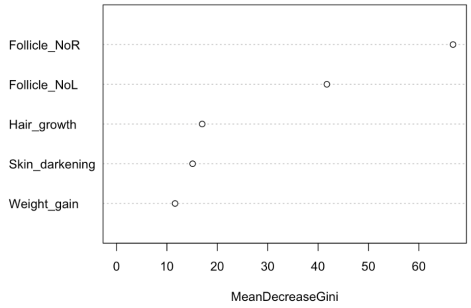


Fig. 14 - Feature importance of second random forest model



Gradient Boosting Machines:

- Model 5 `Gradient Boosting Machines 1` was run against the whole dataset and yield an accuracy of 88.89%
- Model 6 `Gradient Boosting Machines 2` was run against the 6 most influential variables and yield and accuracy of 88.15%

Fig.15 - Summary of first GBM model

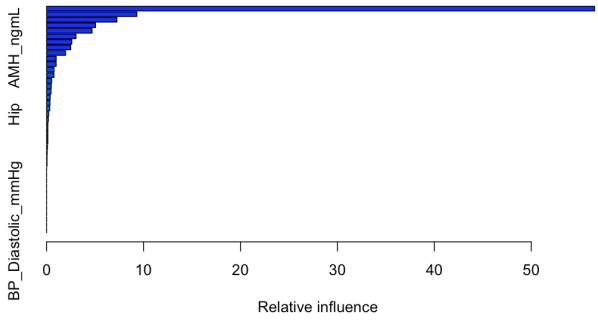
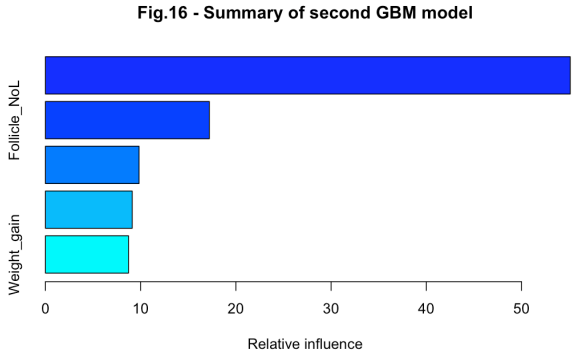


Fig.16 - Summary of second GBM model



Support Vector Machines:

- Model 7 `Support Vector Machines 1` was run against the whole dataset and yield an accuracy of 91.11%
- Model 8 `Support Vector Machines 2` was run against the 6 most influential variables and yield and accuracy of 91.85%

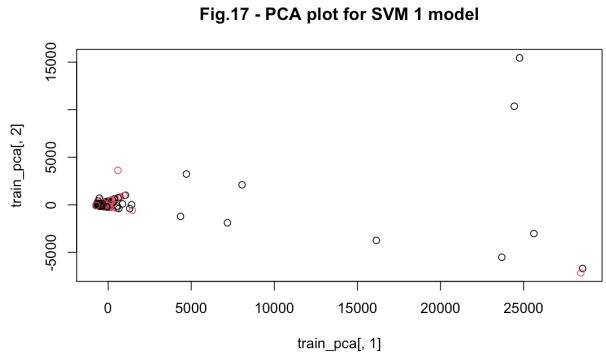


Fig.17 - PCA plot for SVM 1 model

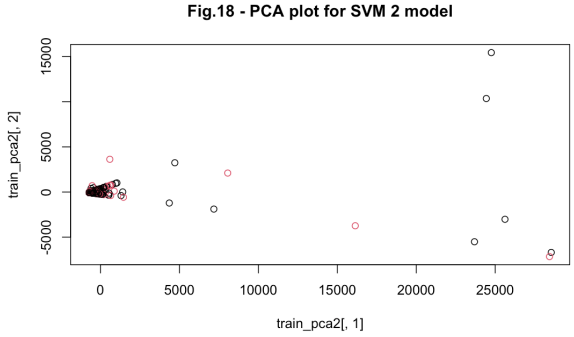
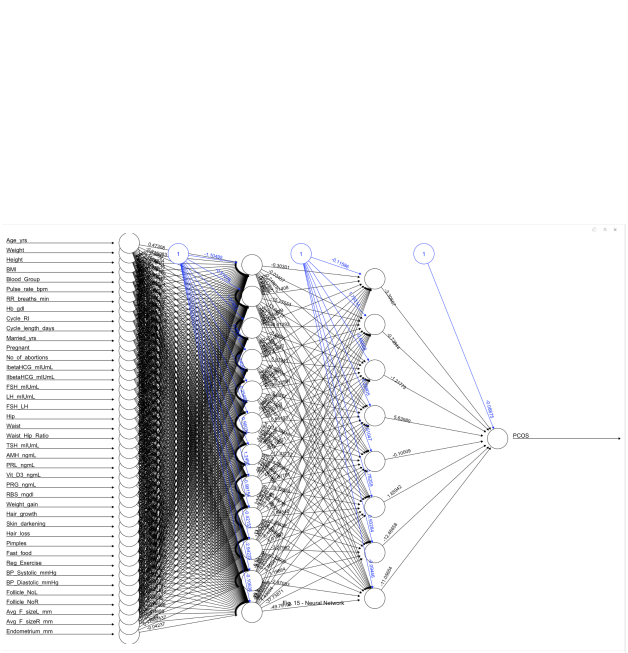


Fig.18 - PCA plot for SVM 2 model

Neural Networks:

- Model 9 `Neural Networks 1` was run against the whole dataset and yield an accuracy of 30.37%
- Model 9.2 `Neural Networks 2` was run against the whole dataset and yield an accuracy of 72.59%
- Model 10 `Neural Networks 3` was run against the 6 most influential variables and yield and accuracy of 30.37%



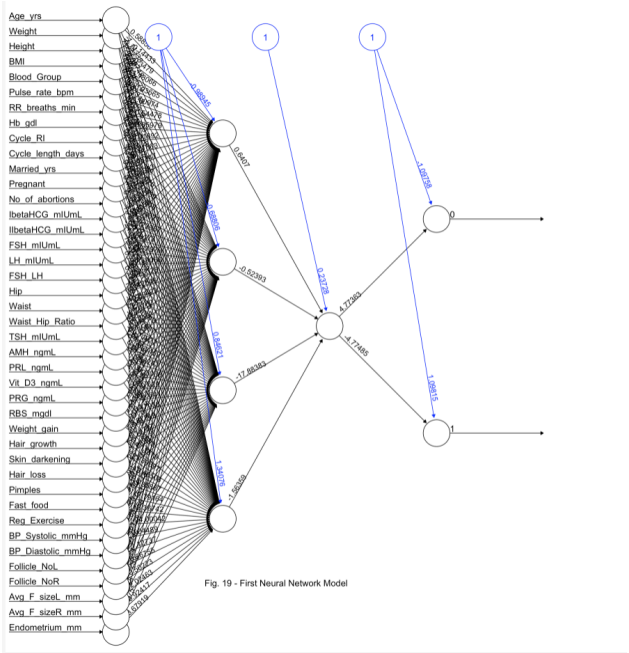


Fig. 19 - First Neural Network Model

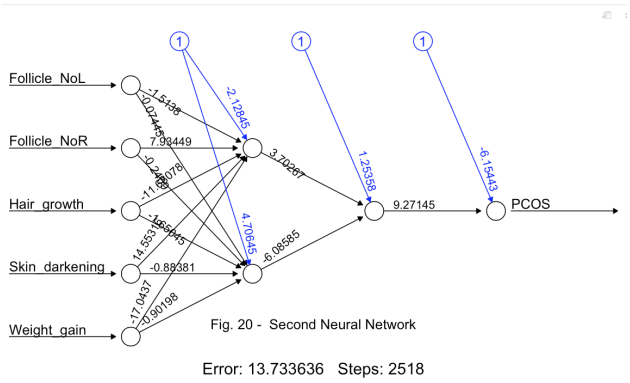
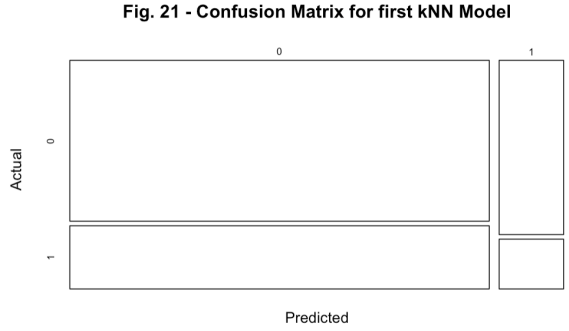
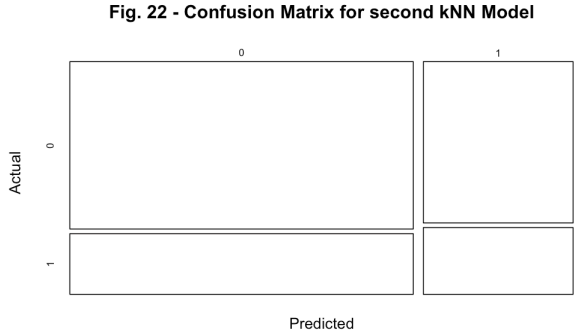


Fig. 20 - Second Neural Network

K-Nearest Neighbors

- Model 11 `k-Nearest Neighbors 1` was run against the whole dataset and yield an accuracy of 65.19%
- Model 12 `k-Nearest Neighbors 2` was run against the 6 most influential variables and yield and accuracy of 88.15%





Overall the top 5 models with the highest accuracies were: Model 2: `Decision Tree 2`, Model 7 and 8: `SVM2` and `SVM 1`, Model 3: `Random Forest 2`, and Model 5: `Gradient Boost Machine 1`.

Table 5. Model Comparison

	Model	Accuracy
2	Decision Tree 2	0.9185185
8	SVM 2	0.9185185
7	SVM 1	0.9111111
4	Random Forest 2	0.9037037
5	GBM 1	0.8814815
13	k-NN 2	0.8814815
1	Decision Tree 1	0.8592593
10	Neural Network 1.2	0.7259259
12	k-NN 1	0.6518519
3	Random Forest 1	0.6296296
6	GBM 2	0.6222222
9	Neural Network 1	0.3037037
11	Neural Network 2	0.3037037

Both Decision Tree 2 and SVM 2 achieved similar high accuracies of approximately 91.85%. This implies that these models were quite successful in making accurate predictions for PCOS diagnosis based on the given data and features. SVM 1 achieved a slightly lower accuracy compared to Decision Tree 2 and SVM 2 but still performed reasonably well at approximately 91.11%. It remains effective in predicting PCOS diagnosis but might be slightly less accurate than the aforementioned models. The second Random Forest model achieved an accuracy of around 90.37%, indicating its capability to correctly classify PCOS and non-PCOS cases with slightly less accuracy than SVM 1 and Decision Tree 2. Both Gradient Boost Machines 1 and k-Nearest 2 models obtained accuracies around 88.15%, which, while lower than the previous models, still demonstrate a moderate ability to predict PCOS diagnosis based on the provided features.

Overall, these accuracies suggest that Decision Tree 2, SVM 2, SVM 1, and Random Forest 2 performed relatively better in distinguishing between PCOS and non-PCOS cases in this dataset, while Gradient Boost Machines 1 and k-Nearest 2 showed somewhat lower but still reasonably acceptable prediction accuracies.

Conclusion:

Initially, I presumed that due to the scarcity of information within the medical domain and the limited availability of datasets for analysis, I held reservations about the potential success in accurately predicting a PCOS diagnosis. Surprisingly, 8 out of the 12 models developed exhibited accuracies surpassing 85%, with only 4 out of the 8 achieving an accuracy exceeding 90%. In comparison to the findings in existing literature utilizing the same Kaggle dataset, the top 5 algorithms I employed showed a slightly lower performance, where these achieved accuracies above 95%. If I had to pick an algorithm to best represent my dataset I'd select `SVM 1` as its accuracy was of 91.11%, a unique percentage not replicated by other algorithms. Conversely, the utilization of Neural Networks in this analysis did not yield satisfactory results, deviating from the expectations set by the literature. This outcome might be attributed to the specific package utilized and the parameter settings chosen for the neural network model.

For the future work, I aim to explore datasets such as NICHDash which are not publicly available and feature a slightly larger and more diverse population within the United States. This exploration will facilitate a more comprehensive assessment of PCOS, enabling a deeper understanding of the condition's nuances and

complexities. I'd also futher delve into additional machine learning methodologies highlighted in the literature reviews. Specifically, I intend to gain more hands-on experience with Support Vector Machines, Neural Networks (NN), and K-Nearest Neighbors (K-NN) algorithms. By combining strategies and considering the constraints posed by limited data, researchers | data scientist can potentially improve the predictive capabilities of machine learning models for diagnosing Polycystic Ovarian Syndrome while maintaining robustness and reliability.

Having subjected the data to various algorithms, I am now able to address the anticipated questions:

1. Are there commonalities women with and without PCOS have that can be easily dismissed as normal?

- Some shared patterns identified through analysis included Body Mass Index (BMI), vital signs, biometric measurements, and specific hormone levels, yet within this dataset, insufficient evidence exists to conclusively dismiss these as unrelated to Polycystic Ovary Syndrome (PCOS).

2. Are there differences for women of different race/ethnic background when it comes to having PCOS? What about women without PCOS?

- These accuracies don't directly address racial or ethnic differences as the dataset only involved women from Kerala, India. Further analysis involving feature exploration or specific subgroup analysis using demographic data might unveil associations or variations among different racial/ethnic groups in PCOS prevalence or features.

3. What is the likelihood of a woman developing PCOS based on her age, ethnicity, and BMI history?

- The accuracies did not explicitly signify predictions regarding probability due to the lack of diverse ethnic backgrounds in the collected records. Moreover, age and BMI were not influential variables across most of the created models in this dataset. Specific models with feature importance or coefficients might offer insights into how age, ethnicity, and BMI contribute to predicting PCOS likelihood but the data would have to include such diversity.

4. Can we predict the risk of insulin resistance, diabetes, and cardiovascular disease in women with PCOS based on their medical history, hormone levels, and lifestyle factors?

- Machine learning models could help predict the risk of insulin resistance, diabetes, and cardiovascular disease in women with PCOS based on available medical history, hormone levels, and lifestyle factors. For this a larger dataset would be needed in order to create specialized modeling to derive precise predictions. Regrettably, my dataset would not be well-suited for conducting such forecasts.

5. Can we predict the likelihood of successful pregnancy outcomes in women with PCOS based on their age, weight, hormone levels, and treatment history?

- Similar to the above, machine learning models can potentially predict the likelihood of successful pregnancy outcomes in women with PCOS based on various factors like age, weight, hormone levels, and treatment history. A more expansive dataset, specialized models, or in-depth analyses could offer more intricate predictive insights, exceeding the capabilities of my current dataset.

6. Can we predict the long-term health outcomes and quality of life of women with PCOS based on their age, lifestyle factors, hormone levels, and treatment history?

- Machine learning models, when trained with extensive data including age, lifestyle factors, hormone levels, and treatment history, might offer predictive insights into long-term health outcomes and quality of life for women with PCOS. However, these models might need additional feature engineering and specialized analyses to offer accurate predictions. I expect that accurately predicting long-term health outcomes, despite an improved dataset, will remain challenging due to the varied presentation of PCOS in women, which poses ongoing diagnostic challenges.

References:

1. Aggarwal, S., & Pandey, K. (2023). Early identification of PCOS with commonly known diseases: Obesity, diabetes, high blood pressure, and heart disease using machine learning techniques. Expert Systems with Applications, 217, 119532. https://doi.org/10.1016/j.eswa.2023.119532

2. Anda, D., & Iyamah, E. (2022, December). Comparative analysis of artificial intelligence in the diagnosis of … ResearchGate. Retrieved April 1, 2023, from https://www.researchgate.net/publication/366320486_Comparative_Analysis_of_Artificial_Intelligence_in_the_Diagnosis_of_Polycystic_Ovary_Syndrome

3. Bartlett, E., & Erlich, L. (2015). Part 3: Dealing with Obstacles — Chapter 5: Polycystic Ovary Syndrome (PCOS). In Feed your fertility: Your guide to cultivating a healthy pregnancy with traditional Chinese medicine, real food, and holistic living (pp. 342–349). essay, Fair Winds Press.

4. Bulsara, J., Patel, P., Soni, A., & Acharya, S. (2021, February 10). A review: Brief insight into polycystic ovarian syndrome. Endocrine and Metabolic Science. Retrieved February 23, 2023, from https://www.sciencedirect.com

5. ELawrence Engmann, Susan Jin, Fangbai Sun, Richard S. Legro, Alex J. Polotsky, Karl R. Hansen, Christos Coutifaris, Michael P. Diamond, Esther Eisenberg, Heping Zhang, Nanette Santoro, C. Bartlebaugh, W. Dodson, S. Estes, C. Gnatuk, J. Ober, R. Brzyski, C. Easton, A. Hernandez, M. Leija, D. Pierce, R. Robinson, A. Awonuga, L. Cedo, A. Cline, K. Collins, S. Krawetz, E. Puscheck, M. Singh, M. Yoscovits, K. Barnhart, K. Lecks, L. Martino, R. Marunich, P. Snyder, R. Alvero, A. Comfort, M. Crow, W. Schlaff, P. Casson, A. Hohmann, S. Mallette, G. Christman, D. Ohl, M. Ringbloom, J. Tang, G. Wright Bates, S. Mason, N. DiMaria, R. Usadi, R. Lucidi, M. Rhea, V. Baker, K. Turner, J. Trussell, D. DelBasso, H. Huang, Y. Li, R. Makuch, P. Patrizio, L. Sakai, L. Scahill, H. Taylor, T. Thomas, S. Tsang, Q. Yan, M. Zhang, D. Haisenleder, C. Lamar, L. DePaolo, D. Guzick, A. Herring, J. Bruce Redmond, M. Thomas, P. Turek, J. Wactawski-Wende, R. Rebar, P. Cato, V. Dukic, V. Lewis, P. Schlegel, F. Witter, Racial and ethnic differences in the polycystic ovary syndrome metabolic phenotype, American Journal of Obstetrics and Gynecology, Volume 216, Issue 5, 2017,Pages 493.e1-493.e13,ISSN 0002-9378, https://doi.org/10.1016/j.ajog.2017.01.003. (https://www.sciencedirect.com/science/article/pii/S0002937817301035)

6. Goodarzi, Carmina, E., & Azziz, R. (2015). DHEA, DHEAS and PCOS. The Journal of Steroid Biochemistry and Molecular Biology, 145, 213–225. https://doi.org/10.1016/j.jsbmb.2014.06.003

7. Hassan, Malik & Mirza, Tabasum. (2020). Comparative Analysis of Machine Learning Algorithms in Diagnosis of Polycystic Ovarian Syndrome. International Journal of Computer Applications. Volume 175. 10.5120/ijca2020920688.

8. Khan, M. J., Ullah, A., & Basit, S. (2019). Genetic Basis of Polycystic Ovary Syndrome (PCOS): Current Perspectives. The application of clinical genetics, 12, 249–260. https://doi.org/

9. Kambale, T., Sawaimul, K. D., & Prakash, S. (2023). A study of hormonal and anthropometric parameters in polycystic ovarian syndrome. Annals of African medicine, 22(1), 112–116. https://doi.org/10.4103/aam.aam_15_22

10. Kavitha, K., Tangudu, N., Sahu, S. R., Narayana, G. V. L., & Anusha, V. (2023). Detection of PCOS using Machine Learning Algorithms with Grid Search CV Optimization. International Journal of Engineering Trends and Technology, 71(7), 201-208. https://doi.org/10.14445/22315381/IJETT-V71I7P219

11. MarchofDimes. (2022). Population of women 15-44 years by age: United States, 2020. March of Dimes | PeriStats. Retrieved February 24, 2023, from https://www.marchofdimes.org

12. Patel, J., & Rai, S. (2018, September). Polycystic ovarian syndrome (PCOS) awareness among young women of central India. ResearchGate. Retrieved March 29, 2023, from https://www.researchgate.net/publication/327566794_Polycystic_ovarian_syndrome_PCOS_awareness_among_young_women_of_central_India

13. Ramanand, S. J., Ghongane, B. B., Ramanand, J. B., Patwardhan, M. H., Ghanghas, R. R., & Jain, S. S. (2013, January). Clinical characteristics of polycystic ovary syndrome in Indian women. Indian journal of endocrinology and metabolism. Retrieved March 18, 2023, from https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3659881/

14. S;, K. M. J. U. A. B. (2019). Genetic basis of polycystic ovary syndrome (PCOS): Current Perspectives. The application of clinical genetics. Retrieved March 18, 2023, from https://pubmed.ncbi.nlm.nih.gov/31920361/

15. S. Nasim, M. S. Almutairi, K. Munir, A. Raza and F. Younas, "A Novel Approach for Polycystic Ovary Syndrome Prediction Using Machine Learning in Bioinformatics," in IEEE Access, vol. 10, pp. 97610-97624, 2022, doi: 10.1109/ACCESS.2022.3205587.

16. Shetty, Disha & Chandrasekaran, Baskaran & Singh, ArulWatson & Oliverraj, Joseph. (2017). Exercise in polycystic ovarian syndrome: An evidence-based review. Saudi Journal of Sports Medicine. 17. 123. 10.4103/sjsm.sjsm_10_17.

17. Tiwari, S., Kane, L., Koundal, D., Jain, A., Alhudhaif, A., Polat, K., Zaguia, A., Alenezi, F., & Althubiti, S. A. (2022). SPOSDS: A smart Polycystic Ovary Syndrome diagnostic system using machine learning. Expert Systems with Applications, 203, 117592. https://doi.org/10.1016/j.eswa.2022.117592

18. Thakre, V., Vedpathak, S., Thakre, K., & Sonawani, S. (2020, December). PCOcare: PCOS detection and prediction using machine learning algorithms. ResearchGate. Retrieved April 1, 2023, from https://www.researchgate.net/publication/348627784_PCOcare_PCOS_Detection_and_Prediction_using_Machine_Learning_Algorithms

19. Thomas, Neetha. (2020). Prediction of polycystic ovarian syndrome with clinical dataset using a novel hybrid data mining classification technique. International journal of advanced research in engineering & technology. 11. 1872-1881,. 10.34218/IJARET.11.11.2020.174.

20. Vikas, B., Anuhya, B. S., Chilla, M., & Sarangi, S. (2018). A Critical Study of Polycystic Ovarian Syndrome (PCOS) Classification Techniques. International Journal of Computational Engineering & Management, 21(4), 1. Retrieved from http://www.ijcem.org/volume21/issue4/IJCEM_2104_01.pdf

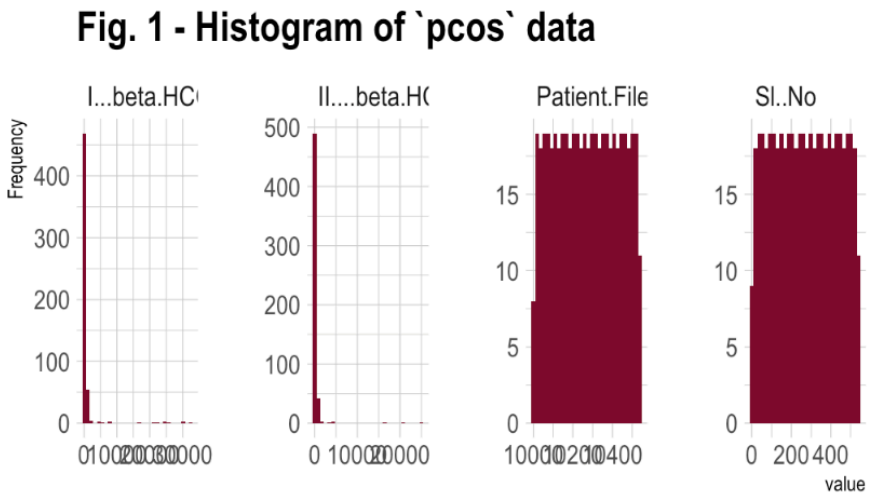
21. Wijeyaratne, C. N., Seneviratne, R.deA., Dahanayake, S., Kumarapeli, V., Palipane, E., Kuruppu, N., Yapa, C., Seneviratne, R.deA., & Balen, A. H. (2011). Phenotype and metabolic profile of South Asian women with polycystic ovary syndrome (PCOS): results of a large database from a specialist Endocrine Clinic. Human reproduction (Oxford, England), 26(1), 202–213. https://doi.org/10.1093/humrep/deq310

22. World female population, 1960-2022. Knoema. (2022). Retrieved February 24, 2023, from https://knoema.com.

23. Wu, J., Chen, X.-Y., Zhang, H., Xiong, L.-D., Lei, H., & Deng, S.-H. (2019). Hyperparameter Optimization for Machine Learning Models Based on Bayesian Optimization. Journal of Electronic Science and Technology, 17(1), 26-40. https://doi.org/10.11989/JEST.1674-862X.80904120

Appendices:

Appendix A - Figures:



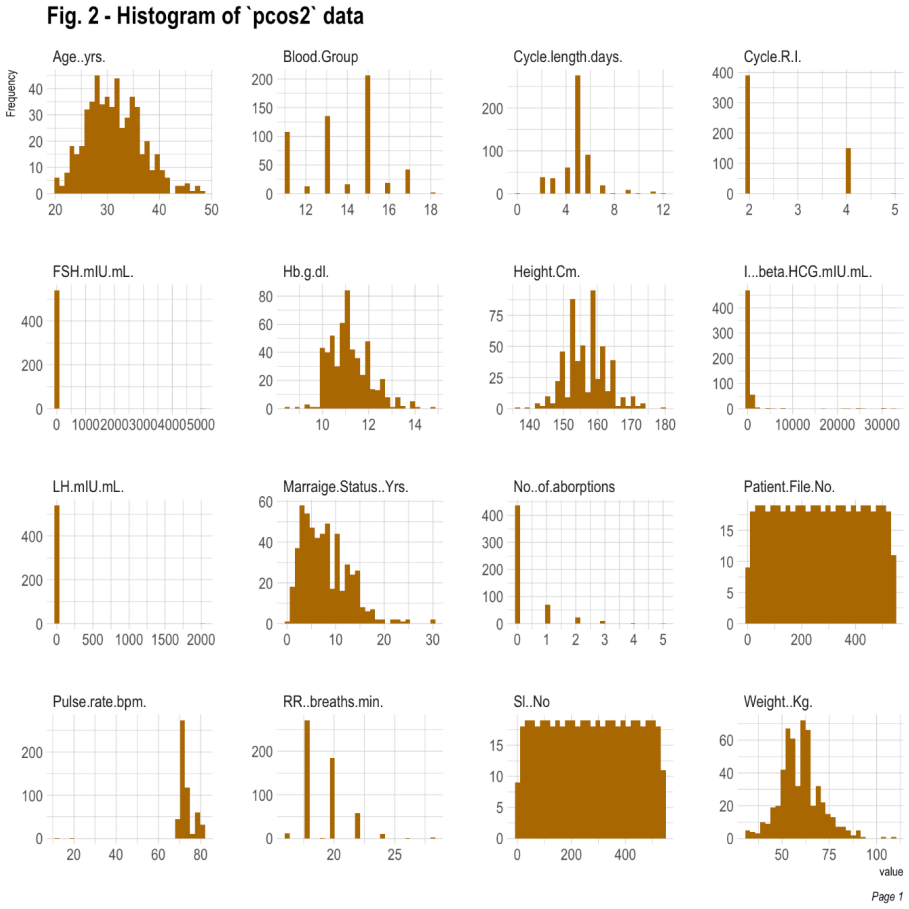


Fig. 2 - Histogram of `pcos2` data

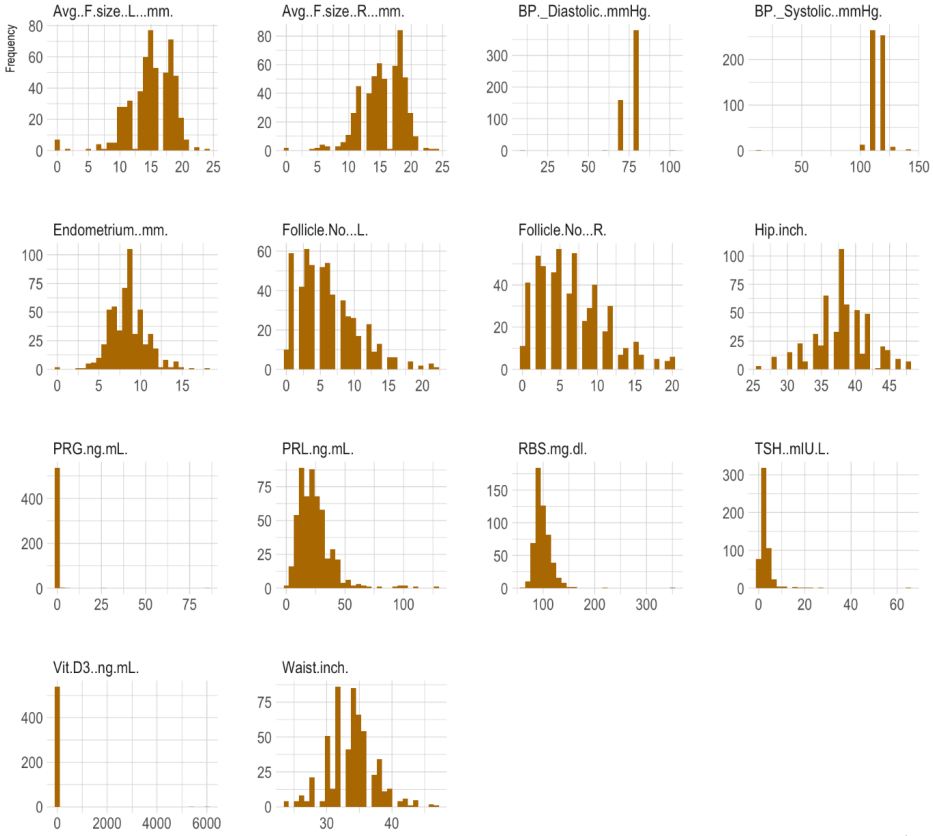


Fig. 3 - Histogram of `pcos_cleaned` data

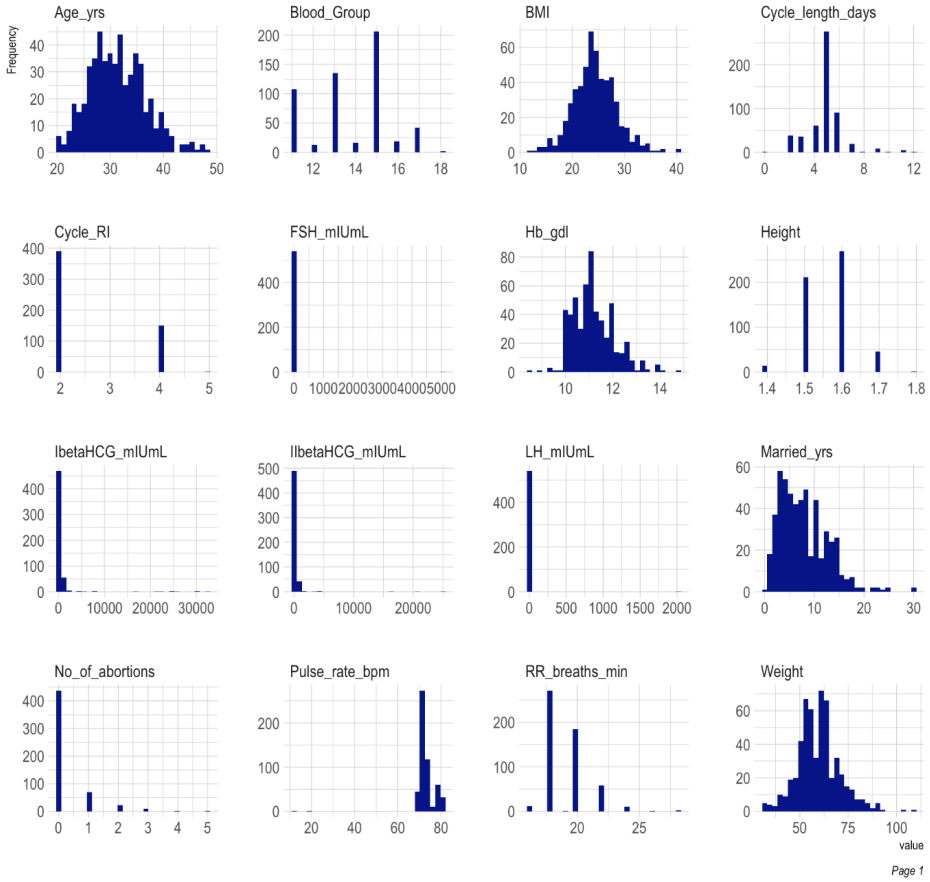


Fig. 3 - Histogram of `pcos_cleaned` data

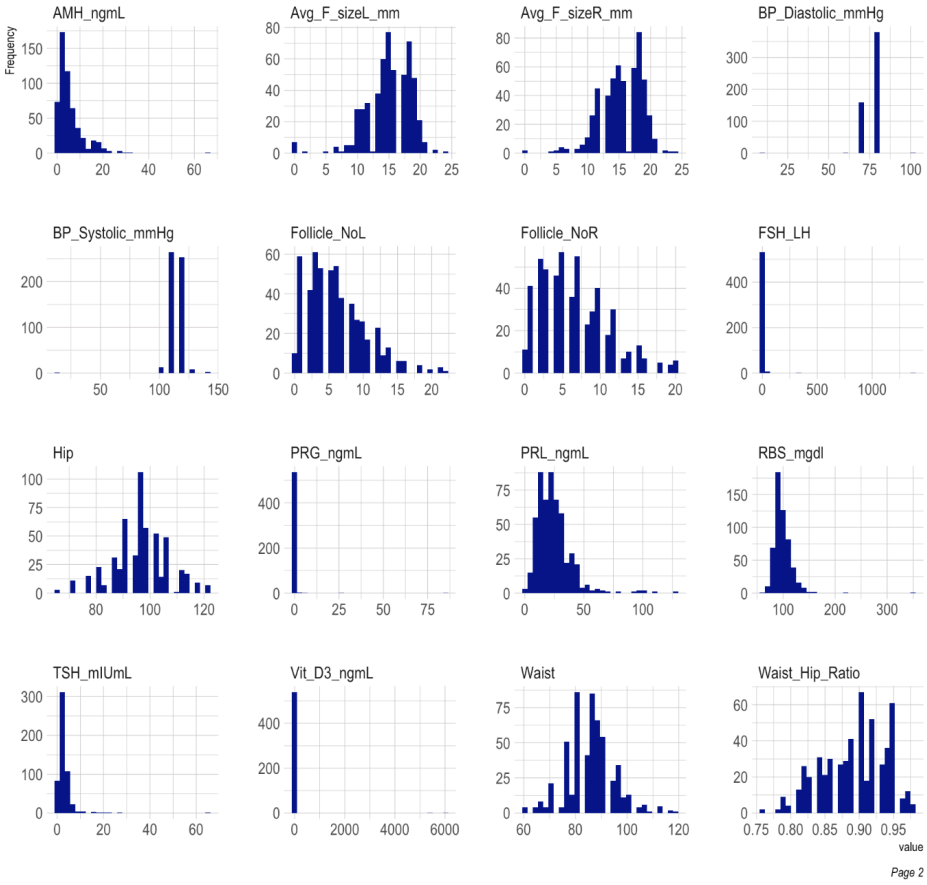


Fig. 3 - Histogram of `pcos_cleaned` data

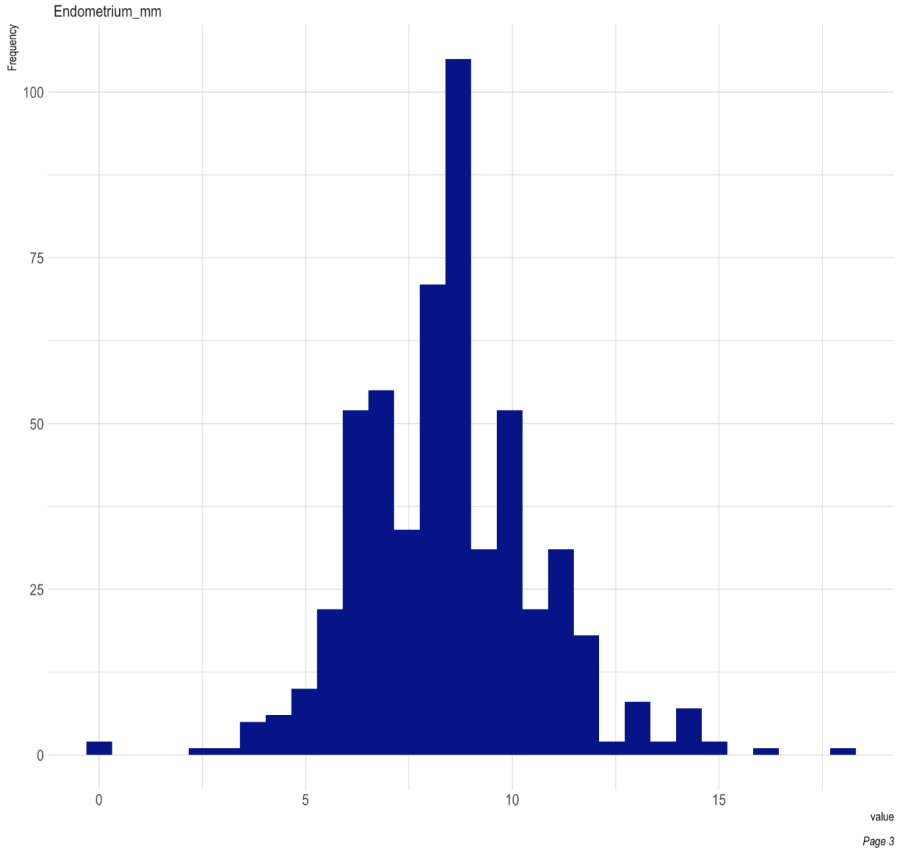
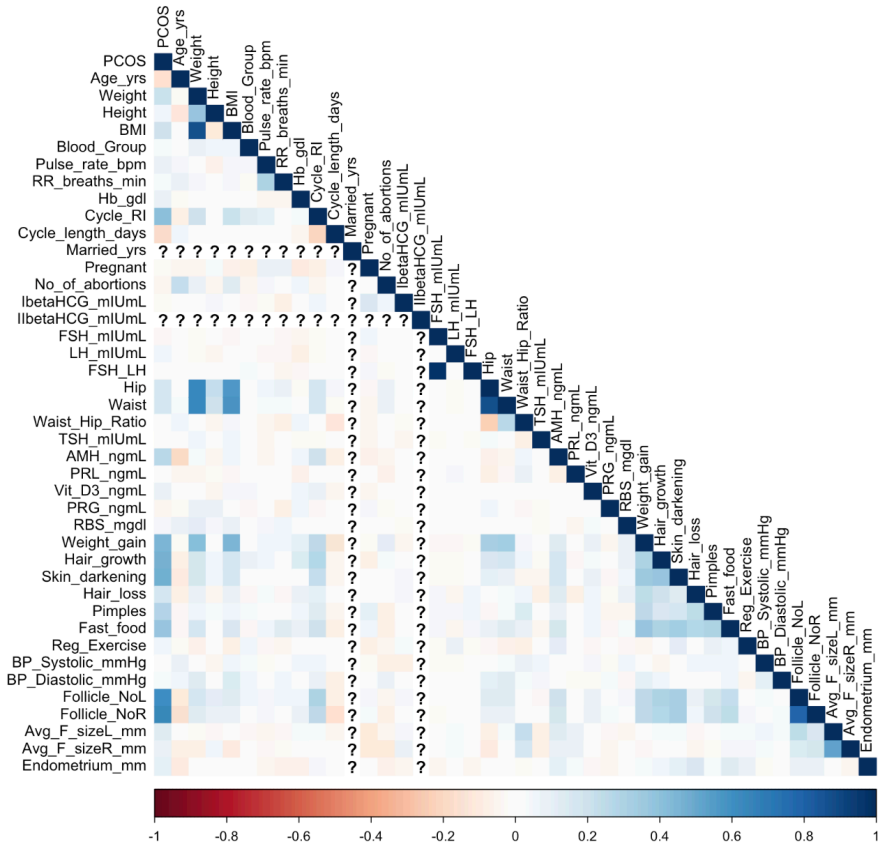


Fig. 4 Correlation plot of `pcos_cleaned` data



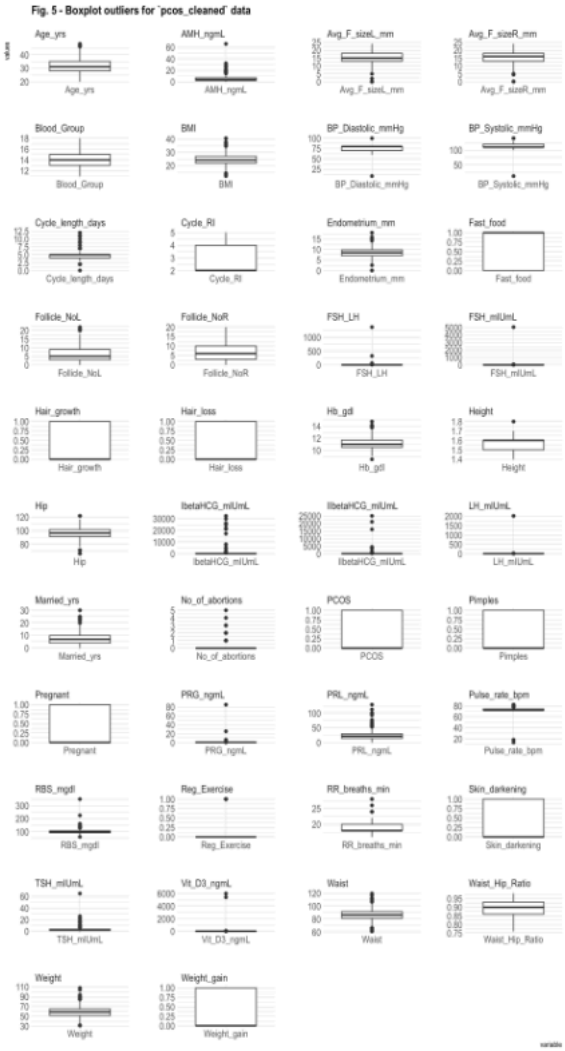


Fig. 6 - Scatterplot of Biometric measures Variables

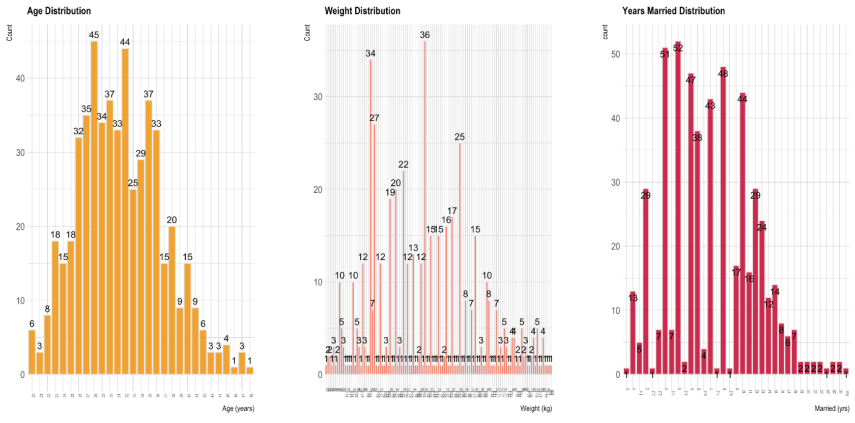


Fig. 7 - Scatterplot of variables with PCOS (Y/N) as factor

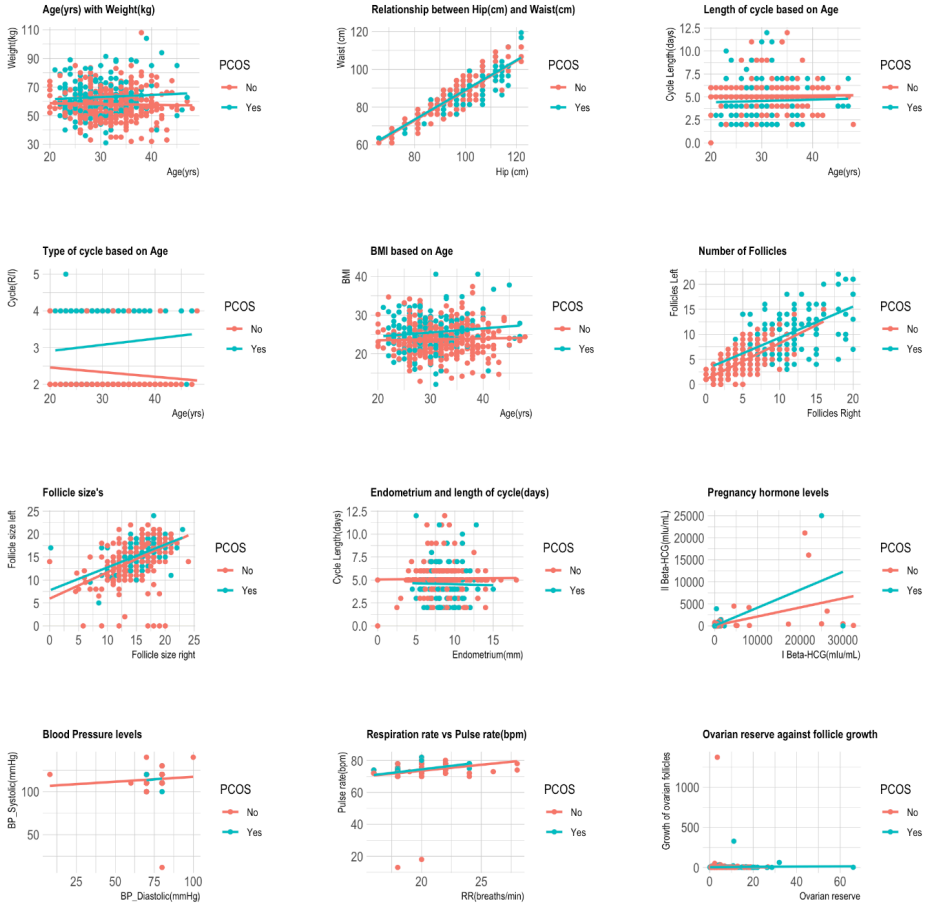


Fig. 8 - Bar charts of Bloodwork variables

Fig. 9 - Bar charts of yes or no variables

Fig. 10 - Bar charts of Biometrics Measures including PCOS (Y/N) as factor

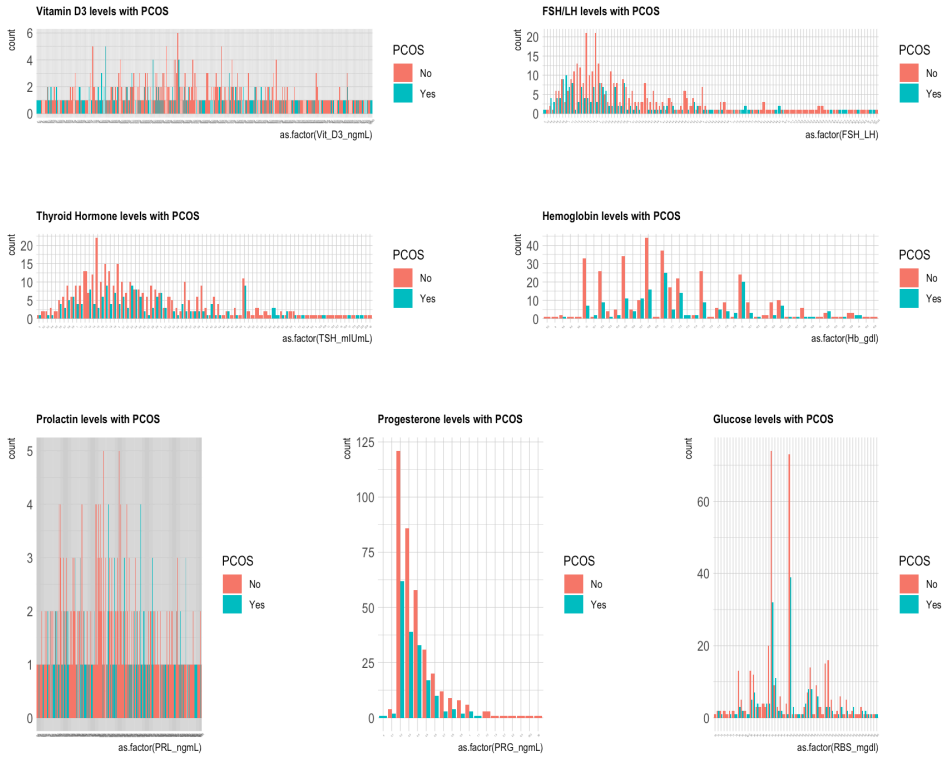


Fig. 11 - Decision Tree with entire dataset

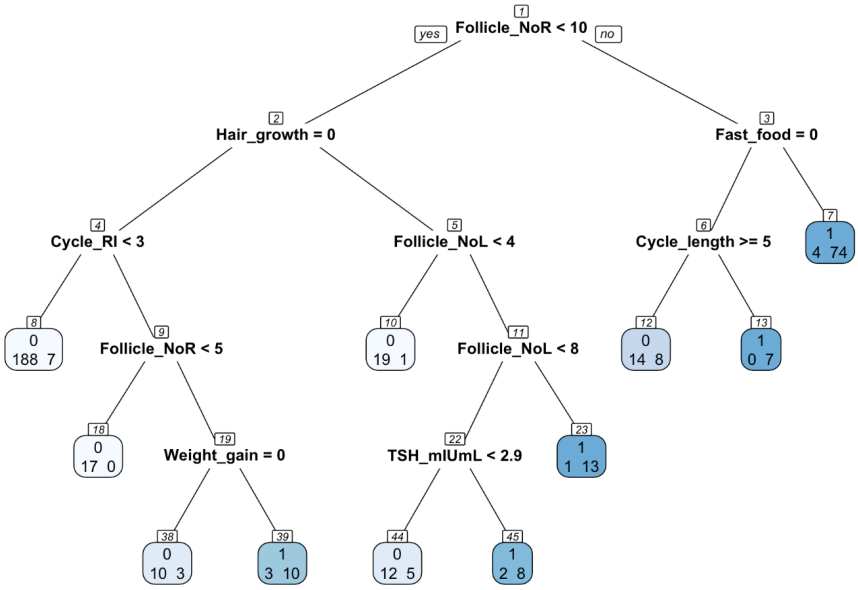


Fig. 12 - Second Decision Tree with 6 variables

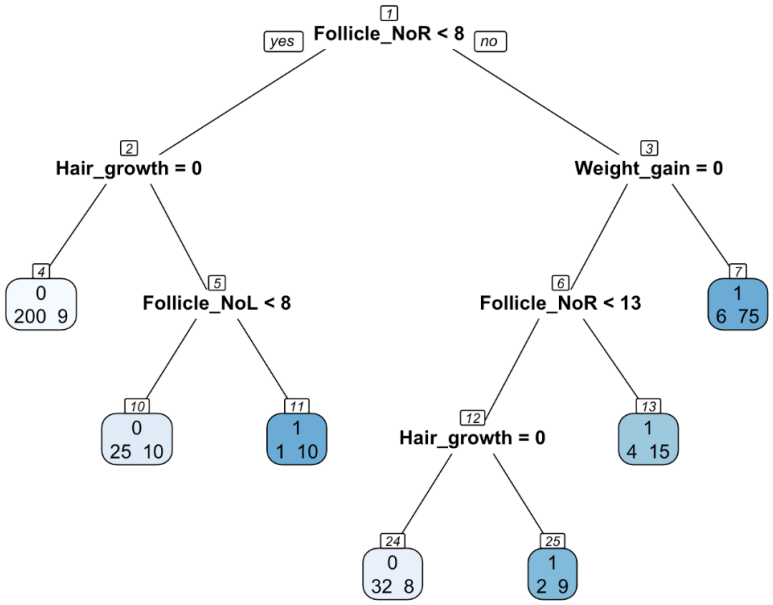


Fig. 13 - Feature importance of first random forest model

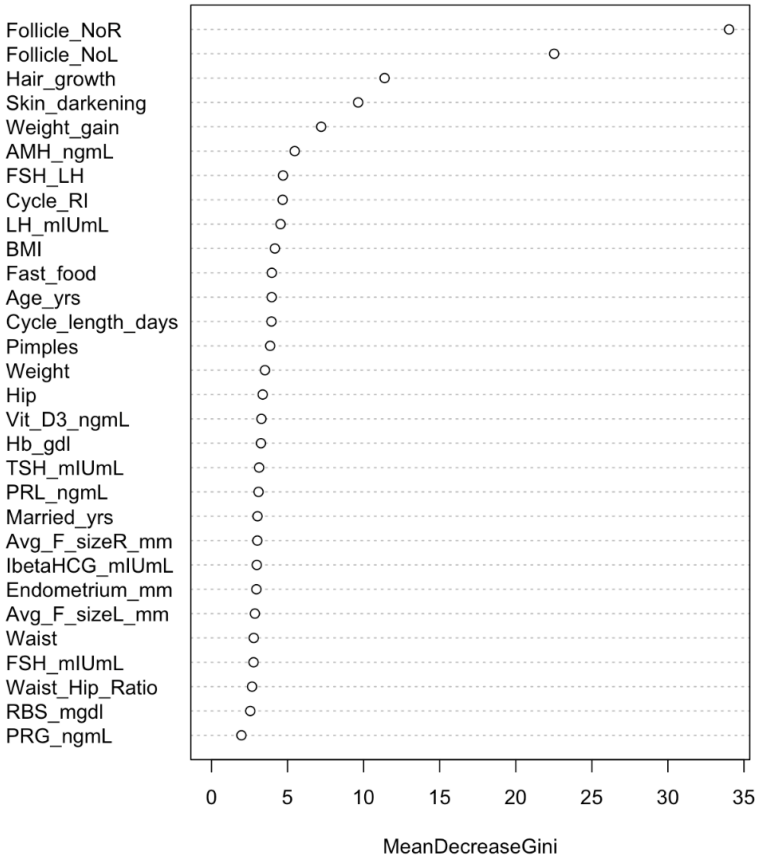


Fig. 14 - Feature importance of second random forest model

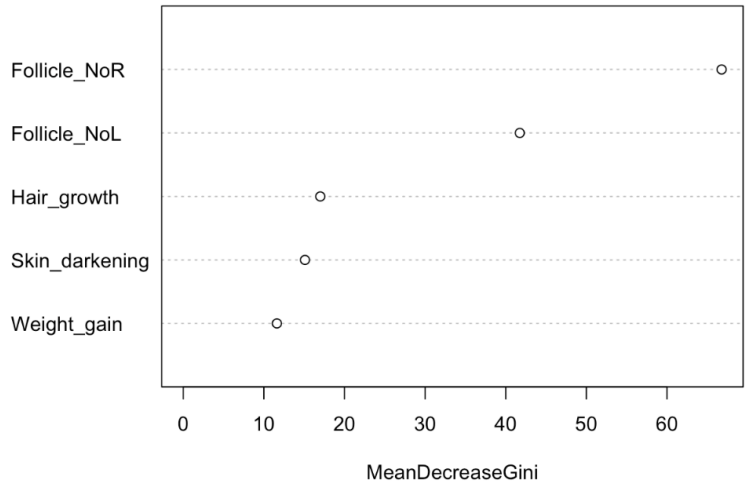


Fig.15 - Summary of first GBM model

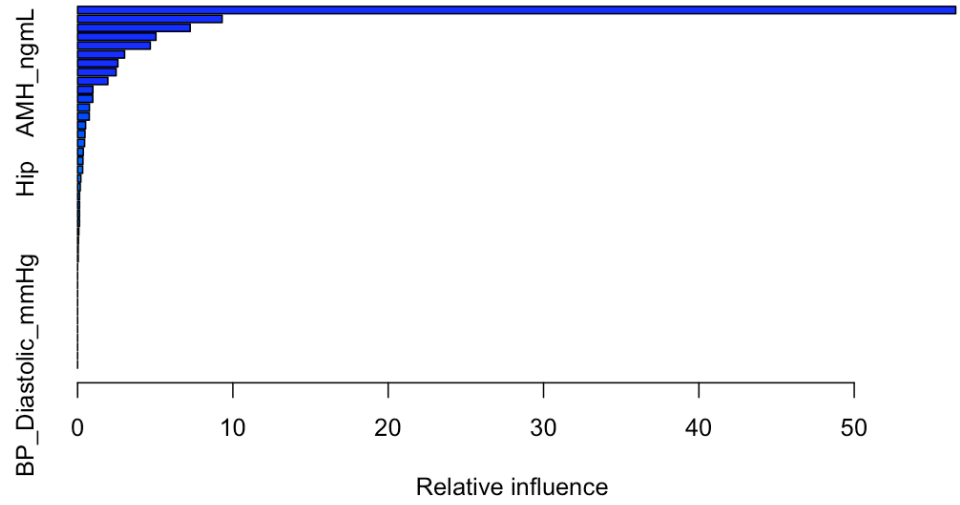


Fig.16 - Summary of second GBM model

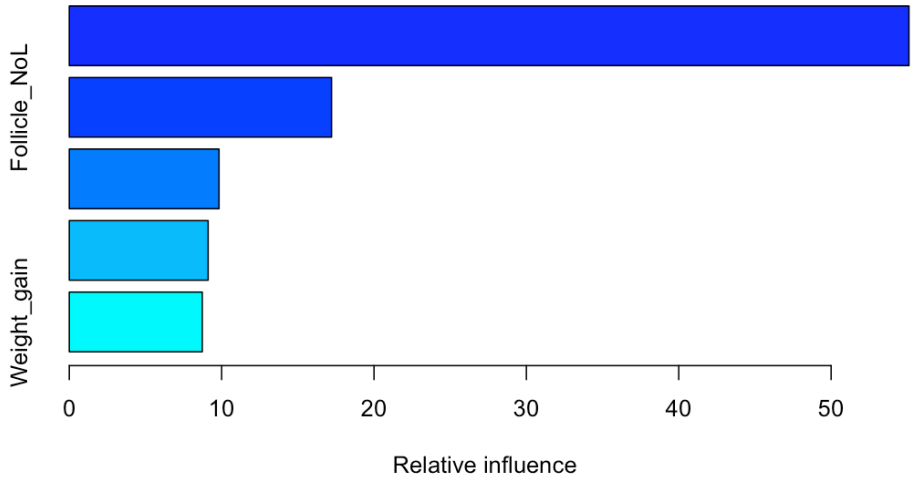


Fig.17 - PCA plot for SVM 1 model

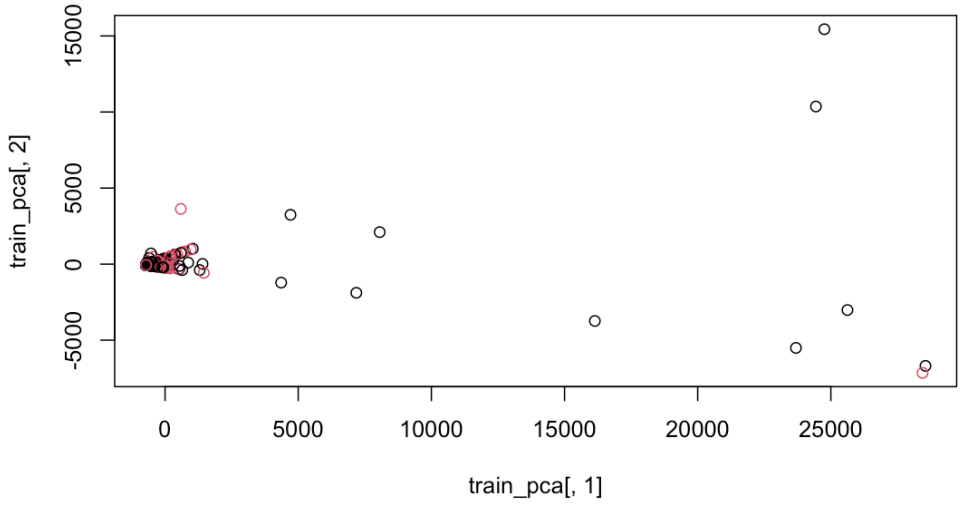
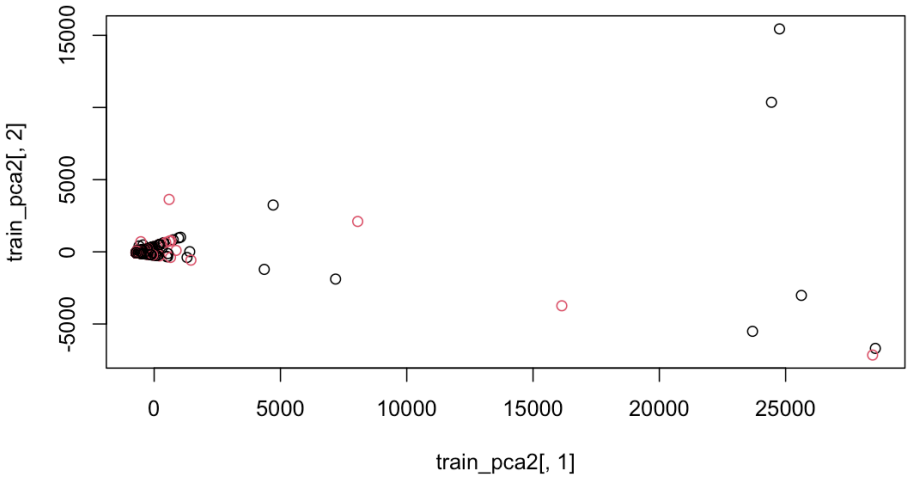


Fig.18 - PCA plot for SVM 2 model



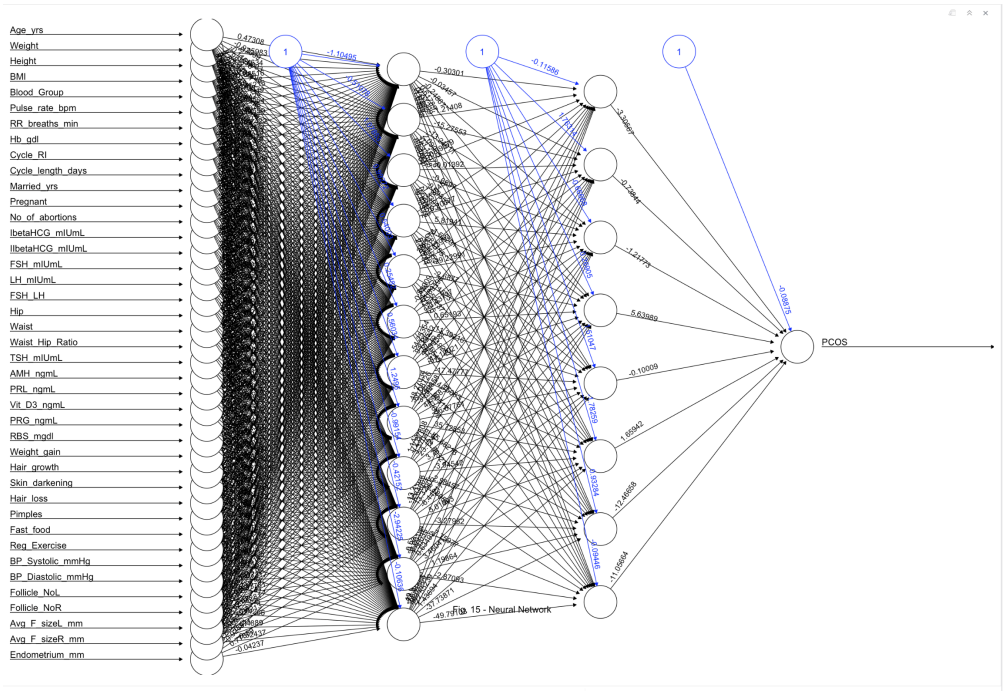


Fig 15 - Neural Network

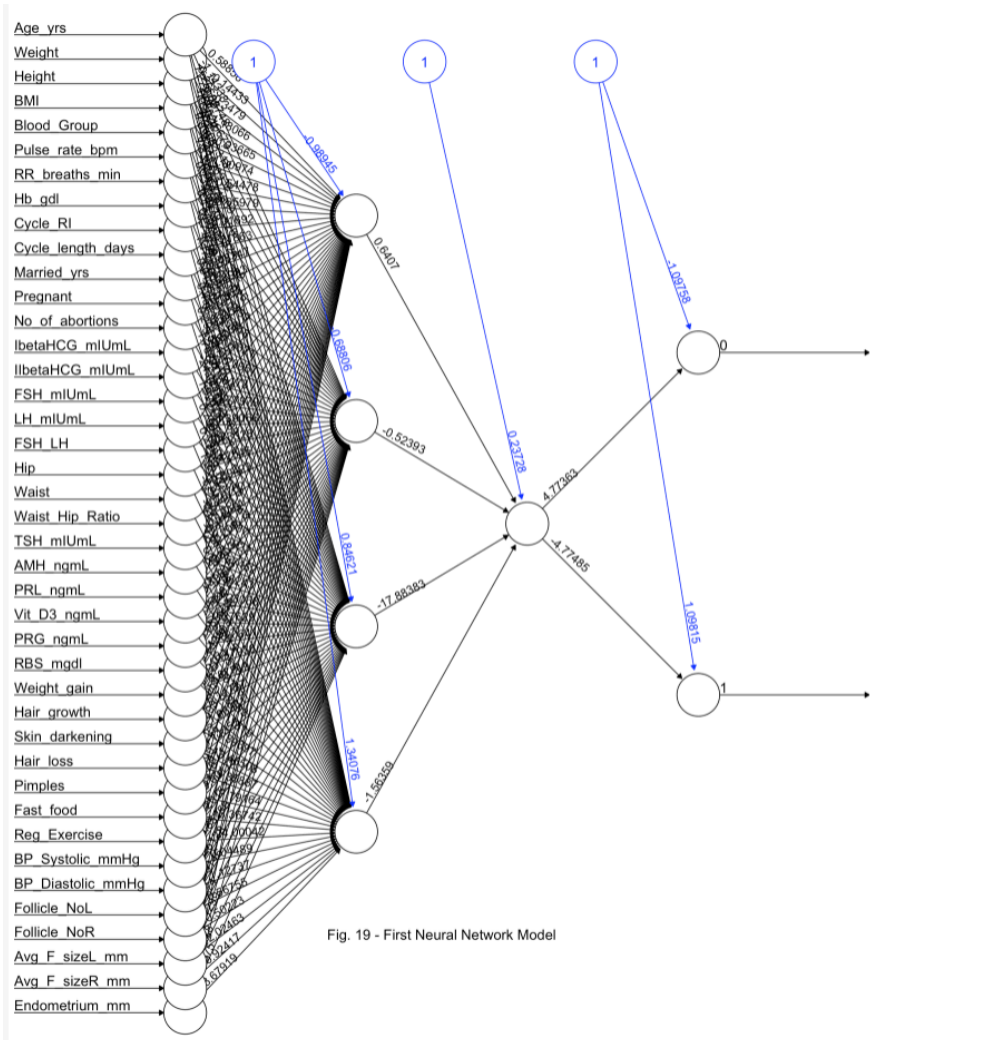


Fig. 19 - First Neural Network Model

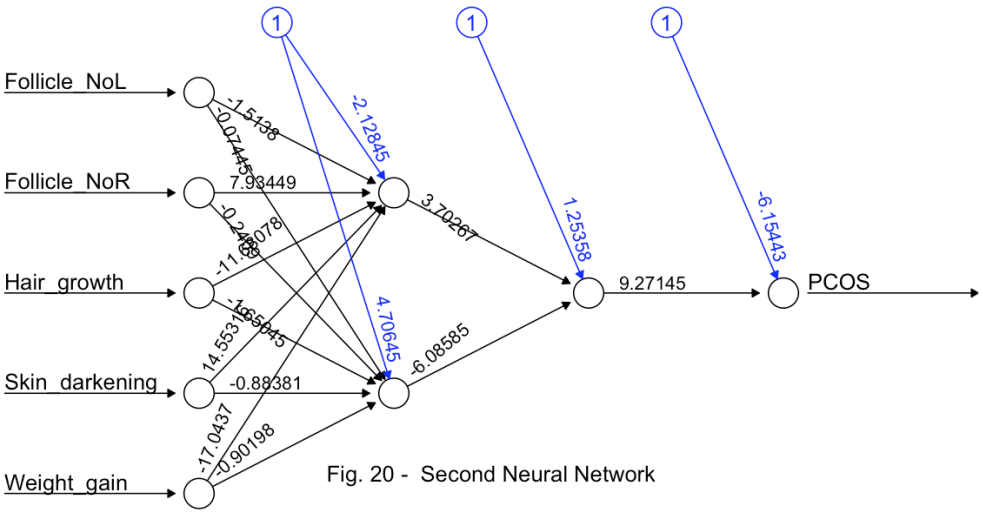


Fig. 20 - Second Neural Network

Error: 13.733636 Steps: 2518

Fig. 21 - Confusion Matrix for first kNN Model

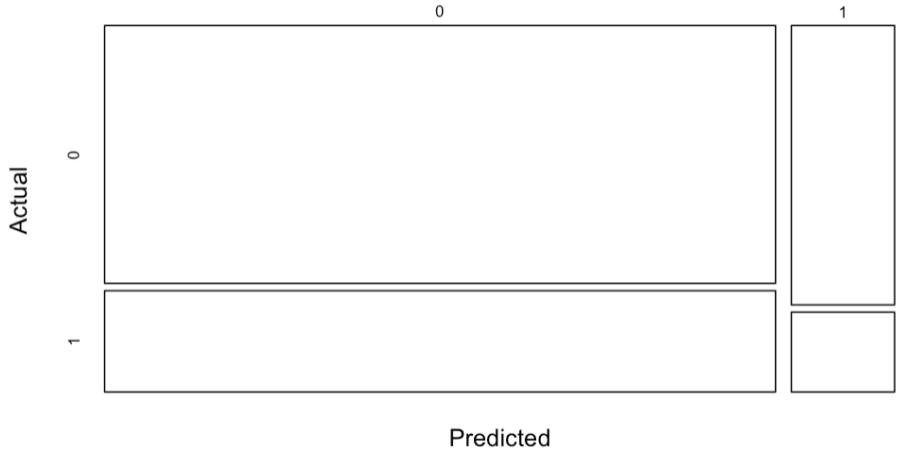
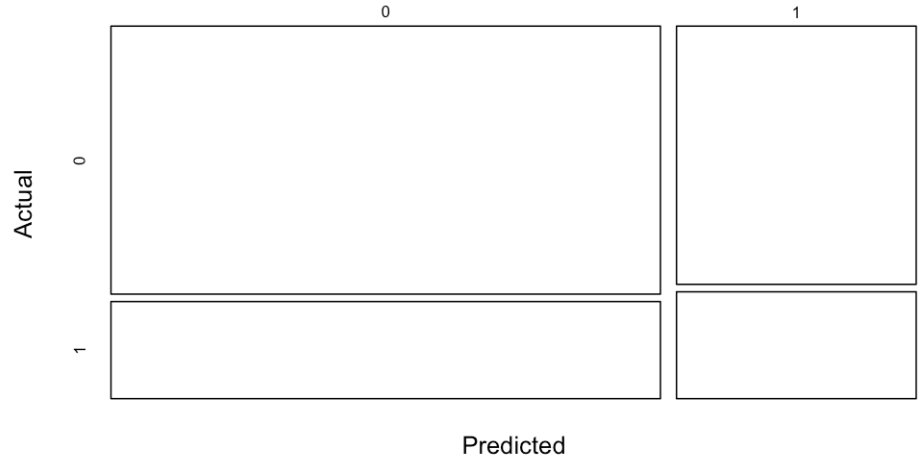


Fig. 22 - Confusion Matrix for second kNN Model



Appendix B - Tables:

Table 1. `pcos` data

Sl..No	Patient.File.No.	PCOS..Y.N.	I...beta.HCG.mIU.mL.	II....beta.HCG.mIU.mL.	AMH.ng.mL.
1	10001	0	1.990	1.990000e+00	2.07
2	10002	0	60.800	1.990000e+00	1.53
3	10003	1	494.080	4.940800e+02	6.63
4	10004	0	1.990	1.990000e+00	1.22
5	10005	0	801.450	8.014500e+02	2.26
6	10006	0	237.970	1.990000e+00	6.74
7	10007	0	1.990	1.990000e+00	3.05
8	10008	0	100.510	1.005100e+02	1.54
9	10009	0	1.990	1.990000e+00	1
10	10010	0	1.990	1.990000e+00	1.61
11	10011	0	158.510	1.585100e+02	4.47

Table 2. `pcos2` data

Sl..No	Patient.File.No.	PCOS..Y.N.	Age..yrs.	Weight..Kg.	Height.Cm.	BMI	Blood.Group	Pulse.rate.bpm.	RR..breaths.min.	Hb.g.dl.	Cycle.R.I.	C
1	1	0	28	44.6	152.000	19.3	15	78	22	10.48	2	
2	2	0	36	65.0	161.500	#NAME?	15	74	20	11.70	2	
3	3	1	33	68.8	165.000	#NAME?	11	72	18	11.80	2	
4	4	0	37	65.0	148.000	#NAME?	13	72	20	12.00	2	
5	5	0	25	52.0	161.000	#NAME?	11	72	18	10.00	2	
6	6	0	36	74.1	165.000	#NAME?	15	78	28	11.20	2	
7	7	0	34	64.0	156.000	#NAME?	11	72	18	10.90	2	
8	8	0	33	58.5	159.000	#NAME?	13	72	20	11.00	2	
9	9	0	32	40.0	158.000	#NAME?	11	72	18	11.80	2	
10	10	0	36	52.0	150.000	#NAME?	15	80	20	10.00	4	
11	11	0	20	71.0	163.000	#NAME?	15	80	20	10.00	2	

Table 3. `pcos_data` merged data

Sl.No	PCOS	Age_yrs	Weight	Height	BMI	Blood_Group	Pulse_rate_bpm	RR_breaths_min	Hb_gdl	Cycle_RI	Cycle_length_days	Married_yrs	P
1	0	28	44.6	152.000	19.3	15	78	22	10.48	2	5	7.0	
2	0	36	65.0	161.500	NA	15	74	20	11.70	2	5	11.0	
3	1	33	68.8	165.000	NA	11	72	18	11.80	2	5	10.0	
4	0	37	65.0	148.000	NA	13	72	20	12.00	2	5	4.0	
5	0	25	52.0	161.000	NA	11	72	18	10.00	2	5	1.0	
6	0	36	74.1	165.000	NA	15	78	28	11.20	2	5	8.0	
7	0	34	64.0	156.000	NA	11	72	18	10.90	2	5	2.0	
8	0	33	58.5	159.000	NA	13	72	20	11.00	2	5	13.0	
9	0	32	40.0	158.000	NA	11	72	18	11.80	2	5	8.0	
10	0	36	52.0	150.000	NA	15	80	20	10.00	4	2	4.0	
11	0	20	71.0	163.000	NA	15	80	20	10.00	2	5	4.0	

Table 4. `pcos_cleaned` dataset

PCOS	Age_yrs	Weight	Height	BMI	Blood_Group	Pulse_rate_bpm	RR_breaths_min	Hb_gdl	Cycle_RI	Cycle_length_days	Married_yrs	Pregnant
0	28	44.6	1.5	19.8	15	78	22	10.5	2	5	7.0	0
0	36	65.0	1.6	25.4	15	74	20	11.7	2	5	11.0	1
1	33	68.8	1.7	23.8	11	72	18	11.8	2	5	10.0	1
0	37	65.0	1.5	28.9	13	72	20	12.0	2	5	4.0	0
0	25	52.0	1.6	20.3	11	72	18	10.0	2	5	1.0	1
0	36	74.1	1.7	25.6	15	78	28	11.2	2	5	8.0	1
0	34	64.0	1.6	25.0	11	72	18	10.9	2	5	2.0	0
0	33	58.5	1.6	22.9	13	72	20	11.0	2	5	13.0	1
0	32	40.0	1.6	15.6	11	72	18	11.8	2	5	8.0	0
0	36	52.0	1.5	23.1	15	80	20	10.0	4	2	4.0	0
0	20	71.0	1.6	27.7	15	80	20	10.0	2	5	4.0	1

Table 5. Model Comparison

	Model	Accuracy
2	Decision Tree 2	0.9185185
8	SVM 2	0.9185185
7	SVM 1	0.9111111
4	Random Forest 2	0.9037037
5	GBM 1	0.8814815
13	k-NN 2	0.8814815
1	Decision Tree 1	0.8592593
10	Neural Network 1.2	0.7259259
12	k-NN 1	0.6518519
3	Random Forest 1	0.6296296
6	GBM 2	0.6222222
9	Neural Network 1	0.3037037
11	Neural Network 2	0.3037037

Appendix C - R Code:

```
# load libraries
library(tidyverse) # data prep
library(DataExplorer) # histograms for datasets
library(skimr) # data prep
library(rpart) # decision tree package
library(rpart.plot) # decision tree display package
library(kableExtra) # kable function for tables
library(knitr) # kable function for table
library(tidyr) # splitting data
library(ggplot2) # graphing
library(hrbrthemes) # chart customization
library(gridExtra) # layering charts
library(stringr) # data prep
```

```r
library(tidymodels) # predictions
library(corrplot) # correlation plot
library(randomForest) # for the random forest
library(caret) # confusion matrix
library("e1071") #svm
library(formattable)
library(corrplot) # correlation plot
library(caret) # confusion matrix
library(neuralnet) # neural network
library(stats) # linear and logistic regression
library(gbm) # generalized boosted models
library(xgboost) # extreme gradient boosting
library(kknn) # weighted k-Nearest neighbors
library(jtools)  # use of summ()
library(patchwork) # ggplot2 multiplot title
library(class) # knn function

# load the dataset from github
pcos <- read.csv("https://raw.githubusercontent.com/letisalba/Data-698/master/Data-Collection-and-Analys
pcos2 <- read.csv("https://raw.githubusercontent.com/letisalba/Data-698/master/Data-Collection-and-Analy

# display the `pcos` dataset
pcos %>%
  kable(caption = "<font color=#000000><b>Table 1.</b>`pcos` data </font>", format = "html", col.names =
  kable_styling(bootstrap_options = c("hover", "condensed"), font_size = 13) %>%
  kableExtra::scroll_box(width = "100%", height = "400px")

# display the `pcos2` dataset
pcos2 %>%
kable(caption = "<font color=#000000><b>Table 2.</b>`pcos2` data </font>", format = "html", col.names =
  kable_styling(bootstrap_options = c("hover", "condensed"), font_size = 13) %>%
  kableExtra::scroll_box(width = "100%", height = "400px")

# summary of the pcos dataset
skim(pcos)

# summary of the pcos2 dataset
skim(pcos2)

DataExplorer::plot_histogram(
  geom_histogram_args = list(alpha = 1, fill = "#7e102c"),
  title = "Fig. 1 - Histogram of `pcos` data",
   data = pcos,
         ggtheme=theme_ipsum())

DataExplorer::plot_histogram(
  geom_histogram_args = list(alpha = 1, fill = "#a86800"),
  title = "Fig. 1 - Histogram of `pcos2` data",
   data = pcos2,
         ggtheme=theme_ipsum())

# change variables to numeric
pcos <- mutate_all(pcos, function(x) as.numeric(as.character(x)))
```

```r
pcos2 <- mutate_all(pcos2, function(x) as.numeric(as.character(x)))

# missing data
colSums(is.na(pcos))

# missing data
colSums(is.na(pcos2))

# removing first two column for `pcos` data
pcos <- dplyr::select(pcos, -c(2:6))

# renaming columns for `pcos` data
pcos <- pcos %>%
  rename("Sl.No" = "Sl..No")

# removing columns not needed for `pcos_infertility` data
pcos2 <- dplyr::select(pcos2, -c(2, 45))

# renaming columns for `pcos_infertility` data
pcos2 <- pcos2 %>%
  rename("Sl.No" = "Sl..No",
         "PCOS" = "PCOS..Y.N.",
         "Age_yrs" = "Age..yrs.",
         "Weight" = "Weight..Kg.",
         "Height" = "Height.Cm.",
         "BMI" = "BMI",
         "Blood_Group" = "Blood.Group",
         "Pulse_rate_bpm" = "Pulse.rate.bpm.",
         "RR_breaths_min" = "RR..breaths.min.",
         "Hb_gdl" = "Hb.g.dl.",
         "Cycle_RI" = "Cycle.R.I.",
         "Cycle_length_days" = "Cycle.length.days.",
         "Married_yrs" = "Marraige.Status..Yrs.",
         "Pregnant" = "Pregnant.Y.N.",
         "No_of_abortions" = "No..of.aborptions",
         "IbetaHCG_mIUmL" = "I...beta.HCG.mIU.mL.",
         "IIbetaHCG_mIUmL" = "II....beta.HCG.mIU.mL.",
         "FSH_mIUmL" = "FSH.mIU.mL.",
         "LH_mIUmL" = "LH.mIU.mL.",
         "FSH_LH" = "FSH.LH",
         "Hip" = "Hip.inch.",
         "Waist" = "Waist.inch.",
         "Waist_Hip_Ratio" = "Waist.Hip.Ratio",
         "TSH_mIUmL" = "TSH..mIU.L.",
         "AMH_ngmL" = "AMH.ng.mL.",
         "PRL_ngmL" = "PRL.ng.mL.",
         "Vit_D3_ngmL" = "Vit.D3..ng.mL.",
         "PRG_ngmL" = "PRG.ng.mL.",
         "RBS_mgdl" = "RBS.mg.dl.",
         "Weight_gain" = "Weight.gain.Y.N.",
         "Hair_growth" = "hair.growth.Y.N.",
         "Skin_darkening" = "Skin.darkening..Y.N.",
         "Hair_loss" = "Hair.loss.Y.N.",
```

```
         "Pimples" = "Pimples.Y.N.",
         "Fast_food" = "Fast.food..Y.N.",
         "Reg_Exercise" = "Reg.Exercise.Y.N.",
         "BP_Systolic_mmHg" = "BP._Systolic..mmHg.",
         "BP_Diastolic_mmHg" = "BP._Diastolic..mmHg.",
         "Follicle_NoL" = "Follicle.No...L.",
         "Follicle_NoR" = "Follicle.No...R.",
         "Avg_F_sizeL_mm" = "Avg..F.size..L...mm.",
         "Avg_F_sizeR_mm" = "Avg..F.size..R...mm.",
         "Endometrium_mm" = "Endometrium..mm.")

# merge data sets
pcos_data <- merge(pcos, pcos2,  by=c("Sl.No"))
# display the merged dataset
pcos_data %>%
kable(caption = "<font color=#000000><b>Table 3.</b>`pcos_data` merged data </font>",
      format = "html", col.names = colnames(pcos_data)) %>%
  kable_styling(bootstrap_options = c("hover", "condensed"), font_size = 13) %>%
  kableExtra::scroll_box(width = "100%", height = "400px")

# convert Height from cm to
pcos_data$"Height" <- round((pcos_data$"Height" * 0.01),1)

# convert hip and waist from inches to cm
pcos_data$"Hip" <- round((pcos_data$"Hip" * 2.54),1)
pcos_data$"Waist" <- round((pcos_data$"Waist" * 2.54),1)

#calculate BMI
pcos_data$"BMI" <- round((pcos_data$"Weight" / pcos_data$"Height"^2), 1)

# calculate waist-hip ratio
pcos_data$"Waist_Hip_Ratio" <- round((pcos_data$"Waist" / pcos_data$"Hip"),2)

# calculate FSH/LH
pcos_data$"FSH_LH" <- round((pcos_data$"FSH_mIUmL"/pcos_data$"LH_mIUmL"),2)

# calculate Married years
pcos_data$"Married(yrs)"[is.na(pcos_data$"Married(yrs)")] <- median(pcos_data$"Married(yrs)",
                                                                   na.rm = T)

# calculate Fast food
pcos_data$"Fast_food"[is.na(pcos_data$"Fast_food")] <- median(pcos_data$"Fast_food",
                                                             na.rm = T)

# calculate
pcos_data$"AMH_ngmL"[is.na(pcos_data$"AMH_ngmL")] <- median(pcos_data$"AMH_ngmL",
                                                           na.rm = T)

# List of variables to round
vars_to_round <- c("Hb_gdl", "Married_yrs", "IbetaHCG_mIUmL", "IbetaHCG_mIUmL",
                   "FSH_mIUmL", "LH_mIUmL", "FSH_LH", "TSH_mIUmL", "AMH_ngmL", "PRL_ngmL",
                   "Vit_D3_ngmL", "PRG_ngmL", "Avg_F_sizeR_mm", "Endometrium_mm")
```

```r
# Rounding the variables to 1 decimal places
pcos_data <- pcos_data %>%
  mutate_at(vars(vars_to_round), ~ round(., digits = 1))


# remove 1st column
pcos_cleaned <- pcos_data[-1]

# display results of cleaned pcos
pcos_cleaned %>%
kable(caption = "<font color=#000000><b>Table 4.</b>`pcos_cleaned` dataset </font>",
      format = "html", col.names = colnames(pcos_cleaned)) %>%
  kable_styling(bootstrap_options = c("hover", "condensed"), font_size = 13) %>%
  kableExtra::scroll_box(width = "100%", height = "400px")

DataExplorer::plot_histogram(
  geom_histogram_args = list(alpha = 1, fill = "dark blue"),
  title = "Fig. 3 - Histogram of `pcos_cleaned` data",
   data = pcos_cleaned,
         ggtheme=theme_ipsum())

# Selecting only the numerical variables for correlation
numerical_data <- pcos_cleaned[, sapply(pcos_cleaned, is.numeric)]

# Calculating the correlation matrix
cor_matrix <- cor(numerical_data)

# Print the correlation matrix
corrplot(cor_matrix, method = "color", type = "lower",
         tl.col = "black", tl.cex = 0.9, title = "Fig. 4 Correlation plot of `pcos_cleaned` data", mar=

# boxplot of the variables with the outlier parameters
pcos_df2 <- pcos_cleaned %>%
                gather(variable, values, 1:dim(pcos_cleaned)[2])
pcos_df2 %>%
  ggplot() +
  geom_boxplot(aes(x = variable, y = values)) +
  facet_wrap(~variable, ncol = 4, scales = "free") +
  ggtitle("Fig. 5 - Boxplot outliers for `pcos_cleaned` data") +
  theme_ipsum() +
    theme(
      plot.title = element_text(size=12)
    )

# Histogram of Age distribution
p1 <- pcos_cleaned %>%
  ggplot(aes(x = as.factor(`Age_yrs`))) +
  geom_histogram(stat = "count", fill = "#F39C12", color = "#e9ecef", alpha = 0.9) +
  geom_text(stat = "count", aes(label = ..count..), position = position_dodge(width = 1), vjust = -0.5,
  labs(title = "Age Distribution", x = "Age (years)", y = "Count") +
  theme_ipsum() +
  theme(
    plot.title = element_text(size = 10), axis.text.x = element_text(angle = 90, vjust = 1, hjust=2, si
```

```r
  )

# Histogram of Weight distribution
p2 <- pcos_cleaned %>%
  ggplot(aes(x = as.factor(`Weight`))) +
  geom_histogram(stat = "count", fill = "#FF5733", color = "#e9ecef", alpha = 0.9) +
  geom_text(stat = "count", aes(label = ..count..), position = position_dodge(width = 1), vjust = -0.5,
    labs(title="Weight Distribution", x ="Weight (kg)", y = "Count") +
    theme_ipsum() +
      theme(
        plot.title = element_text(size=12), axis.text.x = element_text(angle = 90, vjust = 1, hjust=2, s
    )

# Histogram of years married distribution
p3 <- pcos_cleaned %>%
  ggplot(aes(x = as.factor(`Married_yrs`))) +
    geom_histogram(stat="count", show.legend = FALSE, fill = "#C70039", color = "#e9ecef", alpha=0.9) +
    geom_text(aes(label = ..count..), position = position_dodge(width = 1), stat = "count", vjust = 1.5
    labs(title="Years Married Distribution", x ="Married (yrs)", y = "count") +
    theme_ipsum() +
      theme(
        plot.title = element_text(size=12), axis.text.x = element_text(angle = 90, vjust = 1, hjust=2, s
    )

# plot all histograms
ggp_all <- (p1 + p2 + p3) +     # Create grid of plots with title
  plot_annotation(title = "Fig. 6 - Scatterplot of Biometric measures Variables") &
  theme(plot.title = element_text(hjust = 0.5))
ggp_all

# Scatterplot of Age and Weight
p4 <- pcos_cleaned %>%
  ggplot(aes(x=`Age_yrs`, y=`Weight`, color=as.factor(`PCOS`))) +
    geom_point() +
    geom_smooth(method="lm", se=FALSE) +
    scale_colour_discrete("PCOS", labels = c("No",
    "Yes")) +
  labs(title="Age(yrs) with Weight(kg)",
        x ="Age(yrs)", y = "Weight(kg)") +
  theme_ipsum() +
    theme(
      plot.title = element_text(size=10)
    )

# Scatterplot of Hip and Waist
p5 <- pcos_cleaned %>%
  ggplot(aes(x=`Hip`, y=`Waist`, color=as.factor(`PCOS`))) +
    geom_point() +
    geom_smooth(method="lm", se=FALSE) +
    scale_colour_discrete("PCOS", labels = c("No",
    "Yes")) +
  labs(title="Relationship between Hip(cm) and Waist(cm)",
        x ="Hip (cm)", y = "Waist (cm)") +
```

```
  theme_ipsum() +
    theme(
      plot.title = element_text(size=10)
    )

# Scatterplot of Length of Cycle and Age
p6 <- pcos_cleaned %>%
ggplot(aes(x=`Age_yrs`, y=`Cycle_length_days`, color=as.factor(`PCOS`))) +
    geom_point() +
    geom_smooth(method="lm", se=FALSE) +
    scale_colour_discrete("PCOS", labels = c("No",
    "Yes")) +
  labs(title="Length of cycle based on Age",
        x ="Age(yrs)", y = "Cycle Length(days)") +
  theme_ipsum() +
    theme(
      plot.title = element_text(size=10)
    )

# Scatterplot of Type of Cycle and Age
p7 <- pcos_cleaned %>%
ggplot(aes(x=`Age_yrs`, y=`Cycle_RI`, color=as.factor(`PCOS`))) +
    geom_point() +
    geom_smooth(method="lm", se=FALSE) +
    scale_colour_discrete("PCOS", labels = c("No",
    "Yes")) +
  labs(title="Type of cycle based on Age",
        x ="Age(yrs)", y = "Cycle(R/I)") +
  theme_ipsum() +
    theme(
      plot.title = element_text(size=10)
    )

# Scatterplot of BMI and Age
p8 <- pcos_cleaned %>%
ggplot(aes(x=`Age_yrs`, y=BMI, color=as.factor(`PCOS`))) +
    geom_point() +
    geom_smooth(method="lm", se=FALSE) +
    scale_colour_discrete("PCOS", labels = c("No",
    "Yes")) +
  labs(title="BMI based on Age",
        x ="Age(yrs)", y = "BMI") +
  theme_ipsum() +
    theme(
      plot.title = element_text(size=10)
    )

# Scatterplot of number of follicles in left and right ovaries and PCOS
p9 <- pcos_cleaned %>%
ggplot(aes(x=`Follicle_NoR`, y=`Follicle_NoL`, color=as.factor(`PCOS`))) +
    geom_point() +
    geom_smooth(method="lm", se=FALSE) +
    scale_colour_discrete("PCOS", labels = c("No",
```

```
    "Yes")) +
  labs(title="Number of Follicles",
       x ="Follicles Right", y = "Follicles Left") +
  theme_ipsum() +
    theme(
      plot.title = element_text(size=10)
    )

# Scatterplot of follicle size and PCOS
p10 <- pcos_cleaned %>%
ggplot(aes(x=`Avg_F_sizeR_mm`, y=`Avg_F_sizeL_mm`, color=as.factor(`PCOS`))) +
    geom_point() +
    geom_smooth(method="lm", se=FALSE) +
    scale_colour_discrete("PCOS", labels = c("No",
    "Yes")) +
  labs(title="Follicle size's",
       x ="Follicle size right", y = "Follicle size left") +
  theme_ipsum() +
    theme(
      plot.title = element_text(size=10)
    )

# Scatterplot of Endometrium and Length of Cycles
p11 <- pcos_cleaned %>%
ggplot(aes(x=`Endometrium_mm`, y=`Cycle_length_days`, color=as.factor(`PCOS`))) +
    geom_point() +
    geom_smooth(method="lm", se=FALSE) +
    scale_colour_discrete("PCOS", labels = c("No",
    "Yes")) +
  labs(title="Endometrium and length of cycle(days)",
       x ="Endometrium(mm)", y = "Cycle Length(days)") +
  theme_ipsum() +
    theme(
      plot.title = element_text(size=10)
    )

# Scatterplot of pregnancy hormone levels
p12 <- pcos_cleaned %>%
ggplot(aes(x=`IbetaHCG_mIUmL`, y=`IIbetaHCG_mIUmL`, color=as.factor(`PCOS`))) +
    geom_point() +
    geom_smooth(method="lm", se=FALSE) +
    scale_colour_discrete("PCOS", labels = c("No",
    "Yes")) +
  labs(title="Pregnancy hormone levels",
       x ="I Beta-HCG(mIu/mL)", y = "II Beta-HCG(mIu/mL)") +
  theme_ipsum() +
    theme(
      plot.title = element_text(size=10)
    )

# Scatterplot of blood pressure levels against PCOS
p13 <- pcos_cleaned %>%
ggplot(aes(x=`BP_Diastolic_mmHg`, y=`BP_Systolic_mmHg`, color=as.factor(`PCOS`))) +
```

```
    geom_point() +
    geom_smooth(method="lm", se=FALSE) +
    scale_colour_discrete("PCOS", labels = c("No",
    "Yes")) +
  labs(title="Blood Pressure levels",
       x ="BP_Diastolic(mmHg)", y = "BP_Systolic(mmHg)") +
  theme_ipsum() +
    theme(
      plot.title = element_text(size=10)
    )

# Scatterplot of Respiration rate and Pulse rate against PCOS
p14 <- pcos_cleaned %>%
ggplot(aes(x=`RR_breaths_min`, y=`Pulse_rate_bpm`, color=as.factor(`PCOS`))) +
    geom_point() +
    geom_smooth(method="lm", se=FALSE) +
    scale_colour_discrete("PCOS", labels = c("No",
    "Yes")) +
  labs(title="Respiration rate vs Pulse rate(bpm)",
       x ="RR(breaths/min)", y = "Pulse rate(bpm)") +
  theme_ipsum() +
    theme(
      plot.title = element_text(size=10)
    )

# Scatterplot on Ovarian reserve against PCOS
p15 <- pcos_cleaned %>%
ggplot(aes(x=`AMH_ngmL`, y=`FSH_LH`, color=as.factor(`PCOS`))) +
    geom_point() +
    geom_smooth(method="lm", se=FALSE) +
    scale_colour_discrete("PCOS", labels = c("No",
    "Yes")) +
  labs(title="Ovarian reserve against follicle growth",
       x ="Ovarian reserve", y = "Growth of ovarian follicles") +
  theme_ipsum() +
    theme(
      plot.title = element_text(size=10)
    )

# plot all scatterplots
ggp_all2 <- (p4 + p5 + p6) / (p7 + p8 + p9) / (p10 + p11 + p12) / (p13 + p14 + p15) +
  plot_annotation(title = "Fig. 7 - Scatterplot of variables with PCOS yes or no as factor") &
  theme(plot.title = element_text(hjust = 0.5))
ggp_all2

# barchart of PCOS variable
p16 <- pcos_cleaned %>%
  ggplot(aes(x = as.factor(`PCOS`), fill = as.factor(`PCOS`))) +
    geom_bar(position = "dodge") +
    geom_text(aes(label = ..count..), position = position_dodge(width = 1), stat = "count", vjust = 1.5
    scale_fill_discrete(name = "PCOS", labels = c("No", "Yes")) +
    scale_x_discrete(labels=c('No', 'Yes')) +
    labs(title="PCOS", x ="PCOS(Yes or No)", y = "count") +
```

```
   theme_ipsum() +
      theme(
        plot.title = element_text(size=10)
      )

# barchart of Pregnant variable
p17 <- pcos_cleaned %>%
  ggplot(aes(x = as.factor(Pregnant), fill = as.factor(Pregnant))) +
    geom_bar(position = "dodge") +
    geom_text(aes(label = ..count..), position = position_dodge(width = 1), stat = "count", vjust = 1.5
    scale_fill_discrete(name = "Pregnant", labels = c("No", "Yes")) +
    scale_x_discrete(labels=c('No', 'Yes')) +
    labs(title="Pregnant", x ="Pregnant(Yes or No)", y = "count") +
  theme_ipsum() +
    theme(
      plot.title = element_text(size=10)
    )

# barchart of Weight gain variable
p18 <- pcos_cleaned %>%
  ggplot(aes(x = as.factor(Weight_gain), fill = as.factor(Weight_gain))) +
    geom_bar(position = "dodge") +
    geom_text(aes(label = ..count..), position = position_dodge(width = 1), stat = "count", vjust = 1.5
    scale_fill_discrete(name = "Weight Gain", labels = c("No", "Yes")) +
    scale_x_discrete(labels=c('No', 'Yes')) +
    labs(title="Weight Gain", x ="Weight Gain (Yes or No)", y = "count") +
  theme_ipsum() +
    theme(
      plot.title = element_text(size=10)
    )

# barchart of Hair growth variable
p19 <- pcos_cleaned %>%
  ggplot(aes(x = as.factor(Hair_growth), fill = as.factor(Hair_growth))) +
    geom_bar(position = "dodge") +
    geom_text(aes(label = ..count..), position = position_dodge(width = 1), stat = "count", vjust = 1.5
    scale_fill_discrete(name = "Hair Growth", labels = c("No", "Yes")) +
    scale_x_discrete(labels=c('No', 'Yes')) +
    labs(title="Hair Growth", x ="Hair Growth(Yes or No)", y = "count") +
  theme_ipsum() +
    theme(
      plot.title = element_text(size=10)
    )

# barchart of Skin darkening variable
p20 <- pcos_cleaned %>%
  ggplot(aes(x = as.factor(Skin_darkening), fill = as.factor(Skin_darkening))) +
    geom_bar(position = "dodge") +
    geom_text(aes(label = ..count..), position = position_dodge(width = 1), stat = "count", vjust = 1.5
    scale_fill_discrete(name = "Skin Darkening", labels = c("No", "Yes")) +
    scale_x_discrete(labels=c('No', 'Yes')) +
    labs(title="Skin Darkening", x ="Skin Darkening (Yes or No)", y = "count") +
  theme_ipsum() +
```

```
    theme(
      plot.title = element_text(size=10)
    )

# barchart of Hair loss variable
p21 <- pcos_cleaned %>%
  ggplot(aes(x = as.factor(Hair_loss), fill = as.factor(Hair_loss))) +
    geom_bar(position = "dodge") +
    geom_text(aes(label = ..count..), position = position_dodge(width = 1), stat = "count", vjust = 1.5
    scale_fill_discrete(name = "Hair Loss", labels = c("No", "Yes")) +
    scale_x_discrete(labels=c('No', 'Yes')) +
    labs(title="Hair Loss", x ="Hair Loss(Yes or No)", y = "count") +
  theme_ipsum() +
    theme(
      plot.title = element_text(size=10)
    )

# barchart of Pimples variable
p22 <- pcos_cleaned %>%
  ggplot(aes(x = as.factor(Pimples), fill = as.factor(Pimples))) +
    geom_bar(position = "dodge") +
    geom_text(aes(label = ..count..), position = position_dodge(width = 1), stat = "count", vjust = 1.5
    scale_fill_discrete(name = "Pimples", labels = c("No", "Yes")) +
    scale_x_discrete(labels=c('No', 'Yes')) +
    labs(title="Pimples", x ="Pimples(Yes or No)", y = "count") +
  theme_ipsum() +
    theme(
      plot.title = element_text(size=10)
    )

# barchart of Fast food variable
p23 <- pcos_cleaned %>%
  ggplot(aes(x = as.factor(Fast_food), fill = as.factor(Fast_food))) +
    geom_bar(position = "dodge") +
    geom_text(aes(label = ..count..), position = position_dodge(width = 1), stat = "count", vjust = 1.5
    scale_fill_discrete(name = "Fast Food", labels = c("No", "Yes")) +
    scale_x_discrete(labels=c('No', 'Yes')) +
    labs(title="Fast Food", x ="Fast Food(Yes or No)", y = "count") +
  theme_ipsum() +
    theme(
      plot.title = element_text(size=10)
    )

# barchart of Regularly Exercise variable
p24 <- pcos_cleaned %>%
  ggplot(aes(x = as.factor(Reg_Exercise), fill = as.factor(Reg_Exercise))) +
    geom_bar(position = "dodge") +
    geom_text(aes(label = ..count..), position = position_dodge(width = 1), stat = "count", vjust = 1.5
    scale_fill_discrete(name = "Regular Exercise", labels = c("No", "Yes")) +
    scale_x_discrete(labels=c('No', 'Yes')) +
    labs(title="Regularly Exercise", x ="Regular Exercise(Yes or No)", y = "count") +
  theme_ipsum() +
    theme(
```

```
      plot.title = element_text(size=10)
    )

# plot all barcharts
ggp_all3 <- (p16 + p17 + p18) / (p19 + p20 + p21) / (p22 + p23 + p24) +
  plot_annotation(title = "Fig. 8 - Bar charts of Bloodwork variables") &
  theme(plot.title = element_text(hjust = 0.5))
ggp_all3

# barchart of Blood Group variable against PCOS
p25 <- pcos_cleaned %>%
  ggplot(aes(x = Blood_Group, fill = as.factor(`PCOS`))) +
    geom_bar(position = "dodge") +
    ggtitle("Blood Group with PCOS") +
    scale_fill_discrete(name = "PCOS", labels = c("No", "Yes")) +
    theme_ipsum() +
    theme(
      plot.title = element_text(size=10), axis.text.x = element_text(angle = 45)
    )

# barchart of Pregnant variable against PCOS
p26 <- pcos_cleaned %>%
  ggplot(aes(x = as.factor(Pregnant), fill = as.factor(`PCOS`))) +
    geom_bar(position = "dodge") +
    geom_text(aes(label = ..count..), position = position_dodge(width = 1), stat = "count", vjust = 1.5
    ggtitle("Pregnant with PCOS") +
    scale_fill_discrete(name = "PCOS", labels = c("No", "Yes")) +
    scale_x_discrete(labels=c('No', 'Yes')) +
    theme_ipsum() +
    theme(
      plot.title = element_text(size=10), axis.text.x = element_text(angle = 45)
    )

# barchart of Weight gain variable against PCOS
p27 <- pcos_cleaned %>%
  ggplot(aes(x = as.factor(Weight_gain), fill = as.factor(`PCOS`))) +
    geom_bar(position = "dodge") +
    geom_text(aes(label = ..count..), position = position_dodge(width = 1), stat = "count", vjust = 1.5
    ggtitle("Weight gain with PCOS") +
    scale_fill_discrete(name = "PCOS", labels = c("No", "Yes")) +
    scale_x_discrete(labels=c('No','Yes')) +
    theme_ipsum() +
    theme(
      plot.title = element_text(size=10), axis.text.x = element_text(angle = 45)
    )

# barchart of Hair growth variable against PCOS
p28 <- pcos_cleaned %>%
  ggplot(aes(x = as.factor(Hair_growth), fill = as.factor(`PCOS`))) +
    geom_bar(position = "dodge") +
    geom_text(aes(label = ..count..), position = position_dodge(width = 1), stat = "count", vjust = 1.5
    ggtitle("Hair growth with PCOS") +
    scale_fill_discrete(name = "PCOS", labels = c("No", "Yes")) +
```

```
    scale_x_discrete(labels=c('No','Yes')) +
    theme_ipsum() +
    theme(
      plot.title = element_text(size=10), axis.text.x = element_text(angle = 45)
    )

# barchart of Skin darkening variable against PCOS
p29 <- pcos_cleaned %>%
  ggplot(aes(x = as.factor(Skin_darkening), fill = as.factor(`PCOS`))) +
    geom_bar(position = "dodge") +
    geom_text(aes(label = ..count..), position = position_dodge(width = 1), stat = "count", vjust = 1.5
    ggtitle("Skin darkening with PCOS") +
    scale_fill_discrete(name = "PCOS", labels = c("No", "Yes")) +
    scale_x_discrete(labels=c('No','Yes')) +
    theme_ipsum() +
    theme(
      plot.title = element_text(size=10), axis.text.x = element_text(angle = 45)
    )

# barchart of Hair loss variable against PCOS
p30 <- pcos_cleaned %>%
  ggplot(aes(x = as.factor(Hair_loss), fill = as.factor(`PCOS`))) +
    geom_bar(position = "dodge") +
    geom_text(aes(label = ..count..), position = position_dodge(width = 1), stat = "count", vjust = 1.5
    ggtitle("Hair loss with PCOS") +
    scale_fill_discrete(name = "PCOS", labels = c("No", "Yes")) +
    scale_x_discrete(labels=c('No','Yes')) +
    theme_ipsum() +
    theme(
      plot.title = element_text(size=10), axis.text.x = element_text(angle = 45)
    )

# barchart of Pimples variable against PCOS
p31 <- pcos_cleaned %>%
  ggplot(aes(x = as.factor(Pimples), fill = as.factor(`PCOS`))) +
    geom_bar(position = "dodge") +
    geom_text(aes(label = ..count..), position = position_dodge(width = 1), stat = "count", vjust = 1.5
    ggtitle("Pimples with PCOS") +
    scale_fill_discrete(name = "PCOS", labels = c("No", "Yes")) +
    scale_x_discrete(labels=c('No','Yes')) +
    theme_ipsum() +
    theme(
      plot.title = element_text(size=10), axis.text.x = element_text(angle = 45)
    )

# barchart of Fast food variable against PCOS
p32 <- pcos_cleaned %>%
  ggplot(aes(x = as.factor(Fast_food), fill = as.factor(`PCOS`))) +
    geom_bar(position = "dodge") +
    geom_text(aes(label = ..count..), position = position_dodge(width = 1), stat = "count", vjust = 1.5
    ggtitle("Fast food consumption with PCOS") +
    scale_fill_discrete(name = "PCOS", labels = c("No", "Yes")) +
    scale_x_discrete(labels=c('No','Yes')) +
```

```
    theme_ipsum() +
    theme(
      plot.title = element_text(size=10), axis.text.x = element_text(angle = 45)
    )

# barchart of Reg. Exercise variable against PCOS
p33 <- pcos_cleaned %>%
  ggplot(aes(x = as.factor(Reg_Exercise), fill = as.factor(`PCOS`))) +
    geom_bar(position = "dodge") +
    geom_text(aes(label = ..count..), position = position_dodge(width = 1), stat = "count", vjust = 1.5
    ggtitle("Regularly exercises with PCOS") +
    scale_fill_discrete(name = "PCOS", labels = c("No", "Yes")) +
    scale_x_discrete(labels=c('No','Yes')) +
    theme_ipsum() +
    theme(
      plot.title = element_text(size=10), axis.text.x = element_text(angle = 45)
    )

# barchart of Length of Cycle variable against PCOS
p34 <- pcos_cleaned %>%
  ggplot(aes(x = `Cycle_length_days`, fill = as.factor(`PCOS`))) +
    geom_bar(position = "dodge") +
    ggtitle("Cycle length in days with PCOS") +
    scale_fill_discrete(name = "PCOS", labels = c("No", "Yes")) +
    theme_ipsum() +
    theme(
      plot.title = element_text(size=10), axis.text.x = element_text(angle = 45)
    )

# barchart of Number of abortions variable against PCOS
p35 <- pcos_cleaned %>%
  ggplot(aes(x = No_of_abortions, fill = as.factor(`PCOS`))) +
    geom_bar(position = "dodge") +
    ggtitle("Number of abortions with PCOS") +
    scale_fill_discrete(name = "PCOS", labels = c("No", "Yes")) +
    theme_ipsum() +
    theme(
      plot.title = element_text(size=10), axis.text.x = element_text(angle = 45)
    )

# plot all barcharts
ggp_all4 <- (p25 + p26 + p27) / (p28 + p29 + p30) / (p31 + p32 + p33) / (p34 + p35) +
  plot_annotation(title = "Fig. 9 - Bar charts of yes or no variables") &
  theme(plot.title = element_text(hjust = 0.5))
ggp_all4

# Barchart for Vitamin D3 levels with PCOS
p36 <- pcos_cleaned %>%
  ggplot(aes(x = as.factor(`Vit_D3_ngmL`), fill = as.factor(`PCOS`))) +
    geom_bar(position = "dodge") +
    ggtitle("Vitamin D3 levels with PCOS") +
    scale_fill_discrete(name = "PCOS", labels = c("No", "Yes")) +
    theme_ipsum() +
```

```
    theme(
      plot.title = element_text(size=10), axis.text.x = element_text(angle = 45, size = 2)
    )

# Barchart for FSH/LH levels with PCOS
p37 <- pcos_cleaned %>%
  ggplot(aes(x = as.factor(`FSH_LH`), fill = as.factor(`PCOS`))) +
    geom_bar(position = "dodge") +
    ggtitle("FSH/LH levels with PCOS") +
    scale_fill_discrete(name = "PCOS", labels = c("No", "Yes")) +
    theme_ipsum() +
    theme(
      plot.title = element_text(size=10), axis.text.x = element_text(angle = 45, size = 2)
    )


# Barchart for Thyroid Hormone levels with PCOS
p38 <- pcos_cleaned %>%
  ggplot(aes(x = as.factor(`TSH_mIUmL`), fill = as.factor(`PCOS`))) +
    geom_bar(position = "dodge") +
    ggtitle("Thyroid Hormone levels with PCOS") +
    scale_fill_discrete(name = "PCOS", labels = c("No", "Yes")) +
    theme_ipsum() +
    theme(
      plot.title = element_text(size=10), axis.text.x = element_text(angle = 45, size = 2)
    )

# Barchart of hemoglobin levels with PCOS
p39 <- pcos_cleaned %>%
  ggplot(aes(x = as.factor(`Hb_gdl`), fill = as.factor(`PCOS`))) +
    geom_bar(position = "dodge") +
    ggtitle("Hemoglobin levels with PCOS") +
    scale_fill_discrete(name = "PCOS", labels = c("No", "Yes")) +
    theme_ipsum() +
    theme(
      plot.title = element_text(size=10), axis.text.x = element_text(angle = 45, size = 2)
    )

# Barchart of Prolactin levels with PCOS
p40 <- pcos_cleaned %>%
  ggplot(aes(x = as.factor(`PRL_ngmL`), fill = as.factor(`PCOS`))) +
    geom_bar(position = "dodge") +
    ggtitle("Prolactin levels with PCOS") +
    scale_fill_discrete(name = "PCOS", labels = c("No", "Yes")) +
    theme_ipsum() +
    theme(
      plot.title = element_text(size=10), axis.text.x = element_text(angle = 45, size = 2)
    )

# Barchart of Progesterone levels with PCOS
p41 <- pcos_cleaned %>%
  ggplot(aes(x = as.factor(`PRG_ngmL`), fill = as.factor(`PCOS`))) +
    geom_bar(position = "dodge") +
```

```
    ggtitle("Progesterone levels with PCOS") +
    scale_fill_discrete(name = "PCOS", labels = c("No", "Yes")) +
    theme_ipsum() +
    theme(
      plot.title = element_text(size=10), axis.text.x = element_text(angle = 45, size = 2)
    )

# Barchart of Glucose levels with PCOS
p42 <- pcos_cleaned %>%
  ggplot(aes(x = as.factor(`RBS_mgdl`), fill = as.factor(`PCOS`))) +
    geom_bar(position = "dodge") +
    ggtitle("Glucose levels with PCOS") +
    scale_fill_discrete(name = "PCOS", labels = c("No", "Yes")) +
    theme_ipsum() +
    theme(
      plot.title = element_text(size=10), axis.text.x = element_text(angle = 45, size = 2)
    )

# plot barcharts
ggp_all5 <- (p36 + p37 + p38 + p39) / (p40 + p41 + p42)  +
  plot_annotation(title = "Fig. 10 - Bar charts of Yes or No variables") &
  theme(plot.title = element_text(hjust = 0.5))
ggp_all5



# DECISION TREE:

# create some random numbers for reproduction
set.seed(29)

# Cross Validation Set-up
inTrain <- createDataPartition(pcos_cleaned$`PCOS`, p=.75, list = F)
train <- pcos_cleaned[inTrain,]
valid <- pcos_cleaned[-inTrain,]

# create the decision tree
rpart_model <- rpart(`PCOS` ~ ., method = "class", data = train)

# display the decision tree
prp(rpart_model, main = "Fig. 12 - Decision Tree with entire dataset",
    extra=1, faclen=0,  nn=T, box.palette="Blues")

# creating our prediction
rpart_result <- predict(rpart_model,
                        newdata = valid[, !colnames(valid) %in% "PCOS"],
                        type = 'class')

# confusion matrix
confusionMatrix(rpart_result, as.factor(valid$`PCOS`))

# contribution of variables
varImp(rpart_model) %>% kable()
```

```r
# Extract accuracy from the confusion matrix
accuracy_rpart <- confusionMatrix(rpart_result, as.factor(valid$`PCOS`))$overall["Accuracy"]
kable(accuracy_rpart, align = "l")

# creating the second dataset from the original
pcos_cleaned2 <- pcos_cleaned %>%
  select(`PCOS`, `Follicle_NoR`, `Follicle_NoL`, `Weight_gain`,
         `Skin_darkening`, `Hair_growth`)

# create some random number for reproduction
set.seed(28)

# Second Cross Validation Set-up
inTrain2 <- createDataPartition(pcos_cleaned2$`PCOS`, p=.75, list = F)
train2 <- pcos_cleaned2[inTrain2,]
valid2 <- pcos_cleaned2[-inTrain2,]

# create the second decision tree
rpart_model2 <- rpart(`PCOS` ~ ., method = "class", data = train2)

# display the decision tree
prp(rpart_model2, main = "Fig. 13 - Second Decision Tree with 6 variables",
    extra=1, faclen=0,  nn=T, box.palette="Blues")

# creating our prediction
rpart_result2 <- predict(rpart_model2,
                         newdata = valid2[, !colnames(valid2) %in% "PCOS"],
                         type = 'class')

# creating the second confusion matrix
confusionMatrix(rpart_result2, as.factor(valid2$`PCOS`))

# contribution of variables
varImp(rpart_model2) %>% kable()

# Extract accuracy from the confusion matrix
accuracy_rpart2 <- confusionMatrix(rpart_result2,
                                   as.factor(valid2$`PCOS`))$overall["Accuracy"]
kable(accuracy_rpart2, align = "l")

# RANDOM FOREST:

# create some random numbers for reproduction
set.seed(30)

# Cross Validation Set-up
rf_inTrain <- createDataPartition(pcos_cleaned$`PCOS`, p=.75, list = F)
rf_train <- pcos_cleaned[rf_inTrain,]
rf_valid <- pcos_cleaned[-rf_inTrain,]

# check the levels of PCOS using levels()
levels(rf_train$PCOS)
levels(rf_valid$PCOS)
```

```r
# Convert PCOS to factor in rf_train
rf_train$PCOS <- factor(rf_train$PCOS)

# Convert PCOS to factor in rf_valid
rf_valid$PCOS <- factor(rf_valid$PCOS)

# rechecking levels again to ensure no NULL values
levels(rf_train$PCOS)
levels(rf_valid$PCOS)

# explicitly set the levels to match the levels in srf_train.
rf_valid$PCOS <- factor(rf_valid$PCOS, levels = levels(rf_train$PCOS))
levels(rf_valid$PCOS)

# # Check the length of rf_result and rf_valid$PCOS
# length_rf_result <- length(rf_result)
# length_rf_valid <- length(rf_valid$PCOS)
#
# # Print the lengths for comparison
# print(length_rf_result)
# print(length_rf_valid)

#create some random number for reproduction
set.seed(39)

# create random forest model using the training data
rf_model <- randomForest(PCOS~., rf_train)
rf_model

# prediction
rf_result <- predict(rf_model, newdata = valid[, !colnames(valid) %in% "PCOS"])

# Create a confusion matrix
confusionMatrix(data = rf_result, reference = rf_valid$PCOS)

# plot for rf_model
varImpPlot(rf_model)

# table for rf_model variable contribution
varImp(rf_model) %>% kable()

# Extract accuracy from the confusion matrix for the rf_model
accuracy_rf <- confusionMatrix(rf_result, valid$PCOS)$overall["Accuracy"]
accuracy_rf

# create some random numbers for reproduction
set.seed(78)

# Second RF Cross Validation Set-up
rf_inTrain2 <- createDataPartition(pcos_cleaned2$`PCOS`, p=.75, list = F)
rf_train2 <- pcos_cleaned2[rf_inTrain2,]
rf_valid2 <- pcos_cleaned2[-rf_inTrain2,]
```

```r
# check the levels of PCOS using levels()
levels(rf_train2$PCOS)
levels(rf_valid2$PCOS)

# Convert PCOS to factor in rf_train
rf_train2$PCOS <- factor(rf_train2$PCOS)

# Convert PCOS to factor in rf_valid
rf_valid2$PCOS <- factor(rf_valid2$PCOS)

# rechecking levels again to ensure no NULL values
levels(rf_train2$PCOS)
levels(rf_valid2$PCOS)

# explicitly set the levels to match the levels in rf_train.
rf_valid2$PCOS <- factor(rf_valid2$PCOS, levels = levels(rf_train2$PCOS))
levels(rf_valid2$PCOS)

# # Check the length of rf_result and rf_valid$PCOS
# length_rf_result2 <- length(rf_result2)
# length_rf_valid2 <- length(rf_valid2$PCOS)
#
# # Print the lengths for comparison
# print(length_rf_result2)
# print(length_rf_valid2)

# create some random number for reproduction
set.seed(7)

# create the second random forest model using the training data from the third decision tree
rf_model2 <- randomForest(PCOS ~ Follicle_NoR + Follicle_NoL +
                            Weight_gain + Skin_darkening +
                            Hair_growth, data = rf_train2)
rf_model2

# creating the prediction for the third decision tree
rf_result2 <- predict(rf_model2, newdata = rf_valid2[, !colnames(rf_valid2) %in% "PCOS"])

# Convert PCOS column to factor in rf_train2 and rf_valid2
rf_train2$PCOS <- factor(rf_train2$PCOS)
rf_valid2$PCOS <- factor(rf_valid2$PCOS)

# # Check unique levels in rf_result2 and rf_valid2$PCOS
# unique_levels_result <- unique(rf_result2)
# unique_levels_valid <- unique(rf_valid2$PCOS)
#
# # Check if the levels match
# identical(unique_levels_result, unique_levels_valid)
#
# # If levels do not match, manually set levels in rf_result2 to match those in rf_valid2$PCOS
# levels(rf_result2) <- levels(rf_valid2$PCOS)
#
# Convert rf_result2 to factor and align levels with rf_valid2$PCOS
```

```r
rf_result2_factor <- factor(rf_result2, levels = levels(rf_valid2$PCOS))

# Create a confusion matrix
confusionMatrix(data = rf_result2_factor, reference = rf_valid2$PCOS)

# plot for the second rf_model
varImpPlot(rf_model2)

# table for rf_model2 variable contribution
varImp(rf_model2) %>% kable()

# Extract accuracy from the confusion matrix for the rf_model2
accuracy_rf2 <- confusionMatrix(data = rf_result2_factor,
                                reference = rf_valid2$PCOS)$overall["Accuracy"]
accuracy_rf2

# GRADIENT BOOSTING MACHINES:

# Set seed for reproducibility
set.seed(67)

# Train the GBM model
gbm_model <- gbm(`PCOS` ~ ., data = train,
                 distribution = "bernoulli", n.trees = 100,
                 interaction.depth = 4, shrinkage = 0.01,
                 bag.fraction = 0.5)

# Print the summary of the trained model
summary(gbm_model)

# Predict on the validation dataset (assuming 'valid' contains your validation dataset)
gbm_pred <- predict(gbm_model, newdata = valid, type = "response")

# Calculate predicted classes (0 or 1) based on the predicted probabilities
predicted_classes <- ifelse(gbm_pred > 0.5, 1, 0)

# Create confusion matrix
confusionMatrix(data = factor(predicted_classes), reference = factor(valid$`PCOS`))

# Calculate accuracy
gbm_accuracy <- sum(predicted_classes == valid$`PCOS`) / length(valid$`PCOS`)
cat("Accuracy:", gbm_accuracy)

# creating the second dataset from the original
pcos_cleaned3 <- pcos_cleaned %>%
  select(`PCOS`, `Follicle_NoR`, `Follicle_NoL`,
         `Weight_gain`, `Skin_darkening`, `Hair_growth`)

# Set seed for reproducibility
set.seed(68)

# Cross Validation Set-up
inTrain3 <- createDataPartition(pcos_cleaned3$`PCOS`, p=.75, list = F)
```

```r
train3 <- pcos_cleaned3[inTrain3,]
valid3 <- pcos_cleaned3[-inTrain3,]

# Train the GBM model
gbm_model2 <- gbm(`PCOS` ~ ., data = train3,
                  distribution = "bernoulli", n.trees = 100,
                  interaction.depth = 4, shrinkage = 0.01,
                  bag.fraction = 0.5)

# Print the summary of the trained model
summary(gbm_model2)

# Predict on the validation dataset (assuming 'valid' contains your validation dataset)
gbm_pred2 <- predict(gbm_model2, newdata = valid3, type = "response")

# Calculate predicted classes (0 or 1) based on the predicted probabilities
predicted_classes2 <- ifelse(gbm_pred2 > 0.5, 1, 0)

# Create confusion matrix
confusionMatrix(data = factor(predicted_classes2), reference = factor(valid3$`PCOS`))

# Calculate accuracy
gbm_accuracy2 <- sum(predicted_classes2 == valid3$`PCOS`) / length(valid3$`PCOS`)
cat("Accuracy:", gbm_accuracy2)


# SUPPORT VECTOR MACHINES:

# check the levels of PCOS using levels()
levels(train$PCOS)
levels(valid$PCOS)

# Convert PCOS to factor in sum_train
train$PCOS <- factor(train$PCOS)

# Convert PCOS to factor in sum_valid
valid$PCOS <- factor(valid$PCOS)

# rechecking levels again to ensure no NULL values
levels(train$PCOS)
levels(valid$PCOS)

# checking the structure of both valid and train datasets
str(valid)
str(train)

# explicitly set the levels to match the levels in sum_train.
valid$PCOS <- factor(valid$PCOS, levels = levels(train$PCOS))
levels(valid$PCOS)

# # Check the length of sum_result and sum_valid$PCOS
# length_sum_result <- length(sum_result)
# length_sum_valid <- length(sum_valid$PCOS)
```

```r
#
# # Print the lengths for comparison
# print(length_svm_result)
# print(length_svm_valid)

#create some random numbers for reproduction
set.seed(31)

# SVM
svm_model <- svm(PCOS ~ ., train)

# create prediction
svm_result <- predict(svm_model, newdata = valid)

# confusion matrix for svm
confusionMatrix(svm_result, valid$PCOS)

# summary of svm_result
summary(svm_result)

#plot support vector machine
# plot(svm_model, train)

# Using PCA for dimensionality reduction (assuming 'train' has multiple features)
pca_model <- prcomp(train[, -which(names(train) == "PCOS")])  # PCA on features except target
train_pca <- predict(pca_model, train)

# Plotting the reduced dimensions (first two principal components)
plot(train_pca[, 1], train_pca[, 2], col = train$PCOS, main="Fig.17 - PCA plot for SVM 1 model")

#Extract accuracy from the confusion matrix
accuracy_svm <- confusionMatrix(svm_result, as.factor(valid$`PCOS`))$overall["Accuracy"]
accuracy_svm

# create some random numbers for reproduction
set.seed(8)

# Cross Validation Set-up
svm_inTrain2 <- createDataPartition(pcos_cleaned2$PCOS, p=.75, list = FALSE)
svm_train2 <- pcos_cleaned2[svm_inTrain2,]
svm_valid2 <- pcos_cleaned2[-svm_inTrain2,]

# check the levels of PCOS using levels()
levels(svm_train2$PCOS)
levels(svm_valid2$PCOS)

# Convert PCOS to factor in svm_train2
svm_train2$PCOS <- factor(svm_train2$PCOS)

# Convert PCOS to factor in svm_valid2
svm_valid2$PCOS <- factor(svm_valid2$PCOS)

# rechecking levels again to ensure no NULL values
```

```r
levels(svm_train2$PCOS)
levels(svm_valid2$PCOS)

# explicitly set the levels to match the levels in svm_train2
valid$PCOS <- factor(svm_valid2$PCOS, levels = levels(svm_train2$PCOS))
levels(svm_valid2$PCOS)

# # Check the length of svm_result and svm_valid$PCOS
# length_svm_result <- length(svm_result)
# length_svm_valid2 <- length(svm_valid2$PCOS)
#
# # Print the lengths for comparison
# print(length_svm_result)
# print(length_svm_valid2)


# Second SVM
svm_model2 <- svm(PCOS ~ Follicle_NoR + Follicle_NoL +
                    Weight_gain + Skin_darkening +
                    Hair_growth, svm_train2)

# create prediction
svm_result2 <- predict(svm_model2, newdata = svm_valid2)

# confusion matrix for svm_valid2
confusionMatrix(svm_result2, svm_valid2$PCOS)

# summary of the results
summary(svm_result2)

#plot support vector machine
#plot(svm_model2, svm_train2)

# Using PCA for dimensionality reduction (assuming 'train' has multiple features)
pca_model2 <- prcomp(svm_train2[, -which(names(svm_train2) == "PCOS")])  # PCA on features except targe
train_pca2 <- predict(pca_model, train)

# Plotting the reduced dimensions (first two principal components)
plot(train_pca2[, 1], train_pca2[, 2], col = svm_train2$PCOS, main="Fig.18 - PCA plot for SVM 2 model")

#Extract accuracy from the confusion matrix
accuracy_svm2 <- confusionMatrix(svm_result2, svm_valid2$`PCOS`)$overall["Accuracy"]
accuracy_svm2

# NEURAL NETWORKS:

# create some random numbers for reproduction
set.seed(67)

# Cross Validation Set-up
nn_inTrain <- createDataPartition(pcos_cleaned$PCOS, p=.75, list = F)
nn_train <- pcos_cleaned[nn_inTrain,]
nn_valid <- pcos_cleaned[-nn_inTrain,]
```

```r
# set a seed for reproducibility purposes
set.seed(19)

# create the model
# nn_model <- neuralnet(`PCOS`~.,
#                       data = nn_train,
#                       hidden = c(12, 8),  # Specify the number of hidden layers and neurons
#                       linear.output = FALSE,
#                       stepmax = 20000  # Increase the maximum number of iterations
# )

# recreate the first model:
nn_model2 <-  neuralnet(`PCOS`~.,
                        data=train,
                        hidden=c(4,1),
                        linear.output = FALSE,
                        stepmax = 10000  # Increase the maximum number of iterations
)

# create the plot based on the model above
plot(nn_model2, rep = "best")
grid::grid.text("Fig. 19.2 - First Neural Network Model", x = 0.5, y = 0.1)

# make predictions on the test data using a previously trained model
pred <- predict(nn_model2, valid)

# create a vector of labels for the two possible `PCOS(Y/N)` status in the dataset.
labels <- c("0", "1")

# creates a data frame with the column index of the maximum value in each row of the "pred" variable
prediction_label <- data.frame(max.col(pred)) %>%
# use the mutate function to add a new column to the data frame called "pred"
mutate(pred=labels[max.col.pred.]) %>%
select(2) %>%
# convert the data frame to a vector.
unlist()

# print the table
table(valid$`PCOS`, prediction_label)

# set seed for reproduction
set.seed(098)

# checking the accuracy recreation of first model
check <- as.numeric(valid$`PCOS`) == max.col(pred)
nn_accuracy <-  (sum(check)/nrow(valid))
nn_accuracy

# set a seed for reproducibility purposes
set.seed(13)

# create the second model
nn_model2 <-  neuralnet(`PCOS`~Follicle_NoL + Follicle_NoR +
```

```
                          Hair_growth + Skin_darkening + Weight_gain,
                        data=train2,
                        hidden=c(2,1),
                        linear.output = FALSE,
                        stepmax = 10000  # Increase the maximum number of iterations
)

# create the plot based on the model above
plot(nn_model3, rep = "best", main="")
grid::grid.text("Fig. 20 -  Second Neural Network", x = .5, y = .2)

# make predictions on the test data using a previously trained model
pred2 <- predict(nn_model3, valid2)

# create a vector of labels for the two possible `PCOS` status in the dataset.
labels2 <- c("0", "1")

# creates a data frame with the column index of the maximum value in each row of the "pred" variable
prediction_label2 <- data.frame(max.col(pred2)) %>%
# use the mutate function to add a new column to the data frame called "pred"
mutate(pred2=labels2[max.col.pred2.]) %>%
select(2) %>%
# convert the data frame to a vector.
unlist()

# print the table
table(valid2$`PCOS`, prediction_label2)

# checking the accuracy
check2 <- as.numeric(valid2$`PCOS`) == max.col(pred2)
nn_accuracy2 <-  (sum(check2)/nrow(valid2))
nn_accuracy2

# K-NEAREST NEIGHBOR

# Remove rows with missing values from train and valid datasets
train <- train[complete.cases(train), ]
valid <- valid[complete.cases(valid), ]

# set a seed for reproducibility purposes
set.seed(78)

# Set the value of k for kNN
k <- 5  # Change this value as needed

# Fit the kNN model using the training data
knn_model <- knn(train[, -which(names(train) == "PCOS")],
                 valid[, -which(names(valid) == "PCOS")],
                 train$`PCOS`,
                 k = k)

# Create confusion matrix for knn model
conf_matrix <- confusionMatrix(knn_model, valid$`PCOS`)
```

```r
conf_matrix

# Plot confusion matrix
plot(conf_matrix$table, col = conf_matrix$byClass,
     main = "Fig. 21 - Confusion Matrix for first kNN Model",
     xlab = "Predicted",
     ylab = "Actual")

# Calculate accuracy
knn_accuracy <- mean(knn_model == valid$`PCOS`)
knn_accuracy

# Filter and select the desired columns for the new dataset
pcos_cleaned4 <- pcos_cleaned %>%
  select(`PCOS`, `Follicle_NoR`, `Follicle_NoL`, `Weight_gain`,
         `Skin_darkening`, `Hair_growth`)

# Split the data into training and validation sets (if needed)
set.seed(123)  # Set seed for reproducibility
inTrain4 <- createDataPartition(pcos_cleaned4$`PCOS`, p = 0.75, list = FALSE)
train4 <- pcos_cleaned4[inTrain4, ]
valid4 <- pcos_cleaned4[-inTrain4, ]

# Check for missing values and remove them if present
train4 <- train4[complete.cases(train4), ]
valid4 <- valid4[complete.cases(valid4), ]

# Set the value of k for kNN
k <- 5  # Change this value as needed

# Fit the kNN model using the training data
knn_model2 <- knn(train4[, -which(names(train4) == "PCOS")],
                  valid4[, -which(names(valid4) == "PCOS")],
                  train4$`PCOS`,
                  k = k)

# Create confusion matrix for the second model
conf_matrix2 <- confusionMatrix(knn_model2, valid$`PCOS`)
conf_matrix2

# Plot confusion matrix
plot(conf_matrix2$table, col = conf_matrix2$byClass,
     main = "Fig. 22 - Confusion Matrix for second kNN Model",
     xlab = "Predicted",
     ylab = "Actual")

# Calculate accuracy for the new kNN model
knn_accuracy2 <- mean(knn_model2 == valid4$`PCOS`)
knn_accuracy2

# Compare models
model_names <- c("Decision Tree 1","Decision Tree 2",
                 "Random Forest 1", "Random Forest 2",
```

```
                 "GBM 1", "GBM 2",
                 "SVM 1", "SVM 2", "Neural Network 1",
                 "Neural Network 1.2", "Neural Network 2",
                 "k-NN 1", "k-NN 2")
accuracies <- c(0.8592593, 0.9185185,
                0.6296296, 0.9037037,
                0.8814815, 0.6222222,
                0.9111111, 0.9185185,
                0.3037037, 0.7259259, 0.3037037,
                0.6518519, 0.8814815)

# place accuracies in data frame
results <- data.frame(Model = model_names, Accuracy = accuracies)

# order in descending order
results <- results[order(results$Accuracy, decreasing = TRUE), ]

# Display the results
kable(results, caption = "<font color=#000000><b>Table 5. </b>Model Comparison </font>", format = "html
  kable_styling(bootstrap_options = c("hover", "condensed"), font_size = 13) %>%
  kableExtra::scroll_box(width = "100%", height = "500px")
```