Assignment 2a

Leonardo Birindelli

December 29, 2023

Assignment's Theoretical Section

1 Random Variables + Discrete probability

Consider the following pseudo-code:

- Set the parameters: $n = 1000$ (sample size) and $p = 0.001$ (probability of success).
- Generate 5000 uniform random numbers u between 0 and 1.
- For each random number u generated, find the corresponding value x via inverse sampling using a binomial CDF.
- Plot a histogram of the generated x values.
- Overlay the true probability mass functions for both the Binomial and Poisson distributions on the histogram.

What do you expect from this plot? And why?

I expect to obtain a bar graph showing the frequencies of occurrence of the various numerical values generated through the application of inverse sampling using the binomial CDF and characterised by the fact that it shows a decreasing trend to the right, typical of distributions in which there are many events whose probability of success is very low. On the same graph it will also be shown how the poisson distribution can be interpreted as a good approximation of the binomial distribution. This situation is reached because we have a very low probability of success $p = 0.001$ and a large number of trials $n = 1000$ and since we also know that the poisson distribution approaches the binomial distribution if $\lambda = n \times p = 1000 \times 0.001$ where $\lambda = 1$ then the condition is satisfied whereby we expect the two distributions to assume a very similar trend between them.

Note: I also created a python script that compute the previous pseudo-code in order to verify my answer: the concerned file is named `exerciseN1.py`

2 Conditional probability + Bayes theorem. Basics of Sentiment Analysis

The sentiment random variable S is defined as:

$$S = \begin{cases} -1 & \text{if } 0 \leq u \leq 0.4 \text{ (Situation 1: Negative Sentiment)} \\ 1 & \text{if } 0.4 < u \leq 1 \text{ (Situation 2: Positive Sentiment)} \end{cases}$$

Consider the list of all the possible words (countable). The **word random variable** W may assume discrete values w representing the integer position of the word in the list. If a sentence is composed by n words, we observe n realizations of the random variable W. You aim to determine the following probabilities:

$$P(S = 1 \mid W_1 = w_1, \ldots, W_n = w_n) \quad \text{(Positive Sentiment)}$$
$$P(S = -1 \mid W_1 = w_1, \ldots, W_n = w_n) \quad \text{(Negative Sentiment)}$$

for 4 User Reviews of Oppenheimer (I) (2023).

To begin with , I aim to determine the probabilities used to calculate correspondingly the likelihood that a review has a positive sentiment and the one for the opposite case, hence, I obtain, using the multiplication theorem, that:

$$P(S=+1|W_1=w_1,\dots,W_n=w_n)=\frac{P(S=+1)\cdot\prod_{k=1}^{n}P(W_k=w_k|S=+1)}{P(W_1=w_1,\dots,W_n=w_n)}$$

$$P(S=-1|W_1=w_1,\dots,W_n=w_n)=\frac{P(S=-1)\cdot\prod_{k=1}^{n}P(W_k=w_k|S=-1)}{P(W_1=w_1,\dots,W_n=w_n)}$$

Note that the expression $P(W_1=w_1,\dots,W_n=w_n)$ can be also written as :

$$P(S=+1)\cdot\prod_{k=1}^{n}P(W_k=w_k|S=+1)+P(S=-1)\cdot\prod_{k=1}^{n}P(W_k=w_k|S=-1)$$

thanks to the law of probabilities.

Now that I formulate the probabilities I compute the likelihoods of each review in both positive and negative sentiments:

(a) *Horribly boring film with no action, only people talking and talking about really uninteresting things.*
The words to analyse in these case are Horribly","boring","film","action", "people" and "talking" . Hence, the final results are.

$P(S=+1|W_1=\text{Horribly},W_2=\text{boring},W_3=\text{film},W_4=\text{action},W_5=\text{people},$
$W_6=\text{talking},W_7=\text{talking})$

$=\frac{0{,}6\cdot(0{,}1\cdot0{,}1\cdot0{,}6\cdot0{,}7\cdot0{,}6\cdot0{,}2\cdot0{,}2)}{0{,}6\cdot(0{,}1\cdot0{,}1\cdot0{,}6\cdot0{,}7\cdot0{,}6\cdot0{,}2\cdot0{,}2)+0{,}4\cdot(0{,}9\cdot0{,}9\cdot0{,}4\cdot0{,}3\cdot0{,}4\cdot0{,}8\cdot0{,}8)}\approx0.006$

$P(S=-1|W_1=\text{Horribly},W_2=\text{boring},W_3=\text{film},W_4=\text{action},W_5=\text{people},$
$W_6=\text{talking},W_7=\text{talking})$

$=\frac{0{,}4\cdot(0{,}9\cdot0{,}9\cdot0{,}4\cdot0{,}3\cdot0{,}4\cdot0{,}8\cdot0{,}8)}{0{,}6\cdot(0{,}1\cdot0{,}1\cdot0{,}6\cdot0{,}7\cdot0{,}6\cdot0{,}2\cdot0{,}2)+0{,}4\cdot(0{,}9\cdot0{,}9\cdot0{,}4\cdot0{,}3\cdot0{,}4\cdot0{,}8\cdot0{,}8)}\approx0.994$

(b) *There's a terrible soundtrack of melodramatic music to make sure you know how you're supposed to feel, frequent random booming and rumbling to make your liver quiver, and screen-filling explosions that come out of nowhere and seem designed only to startle - sort of a science equivalent of jump scares.*

The words to analyse in this case are "terrible","soundtrack","melodramatic","music", "feel","frequent", "random","booming","rumbling","liver","quiver","explosions","science","equivalent","jump","scares". Therefore, the final result is.

$P(S=+1|W_1=\text{terrible},W_2=\text{soundtrack},W_3=\text{melodramatic},W_4=\text{music},W_5=\text{feel},$
$W_6=\text{frequent},W_7=\text{random},W_8=\text{booming},W_9=\text{rumbling},W_{10}=\text{liver},W_{11}=\text{quiver},$
$W_{12}=\text{explosions},W_{13}=\text{science},W_{14}=\text{equivalent},W_{15}=\text{jump},W_{16}=\text{scares})$
$=\frac{P_{positive}}{P_{positive}+P_{negative}}\approx0.007$

$P(S=-1|W_1=\text{terrible},W_2=\text{soundtrack},W_3=\text{melodramatic},W_4=\text{music},W_5=\text{feel},$
$W_6=\text{frequent},W_7=\text{random},W_8=\text{booming},W_9=\text{rumbling},W_{10}=\text{liver},W_{11}=\text{quiver},$
$W_{12}=\text{explosions},W_{13}=\text{science},W_{14}=\text{equivalent},W_{15}=\text{jump},W_{16}=\text{scares})$
$=\frac{P_{negative}}{P_{positive}+P_{negative}}\approx0.993$

Where
Ppositive $= 0,6 \cdot (0,2 \cdot 0,8 \cdot 0,2 \cdot 0,5 \cdot 0,8 \cdot 0,6 \cdot 0,7 \cdot 0,3 \cdot 0.3 \cdot 0.1 \cdot 0.1 \cdot 0.3 \cdot 0.6 \cdot 0.6 \cdot 0.6 \cdot 0.3)$

Pnegative $= 0,4 \cdot (0,8 \cdot 0,2 \cdot 0,8 \cdot 0,5 \cdot 0,2 \cdot 0,4 \cdot 0,3 \cdot 0,7 \cdot 0,7 \cdot 0,9 \cdot 0,9 \cdot 0,7 \cdot 0,4 \cdot 0,4 \cdot 0,4 \cdot 0,7)$

Note : The variables Ppositive and Pnegative respectively represent the probability that a review is positive and that a review is negative. I decided to use two variables because, in the final representation of the used formula, the calculations were not fit in the pdf very well and the computations appeared quite chaotic. This approach is only use to clarify the document presentation .

(c) *Half of this movie is something related with quantum physic and the rest all about law and politic. However if you aren't into those things, you can still understand and enjoy the movies.*

The words to analyse in this case are "half","related","quantum","physic", "rest","law","politic", "enjoy". Therefore, the final result is.

$P(S = +1|W_1 = \text{half}, W_2 = \text{related}, W_3 = \text{quantum}, W_4 = \text{physic}, W_5 = \text{rest}, W_6 = \text{law},$
$W_7 = \text{politic}, W_8 = \text{enjoy})$
$= \frac{0.6 \cdot (0.5 \cdot 0.7 \cdot 0.8 \cdot 0.8 \cdot 0.5 \cdot 0.2 \cdot 0.2 \cdot 0.8)}{0.6 \cdot (0.5 \cdot 0.7 \cdot 0.8 \cdot 0.8 \cdot 0.5 \cdot 0.2 \cdot 0.2 \cdot 0.8) + 0.4 \cdot (0.5 \cdot 0.3 \cdot 0.2 \cdot 0.2 \cdot 0.5 \cdot 0.8 \cdot 0.8 \cdot 0.2)} \approx 0.93$

$P(S = -1|W_1 = \text{half}, W_2 = \text{related}, W_3 = \text{quantum}, W_4 = \text{physic}, W_5 = \text{rest}, W_6 = \text{law},$
$W_7 = \text{politic}, W_8 = \text{enjoy})$
$= \frac{0.4 \cdot (0.5 \cdot 0.3 \cdot 0.2 \cdot 0.2 \cdot 0.5 \cdot 0.8 \cdot 0.8 \cdot 0.2)}{0.6 \cdot (0.5 \cdot 0.7 \cdot 0.8 \cdot 0.8 \cdot 0.5 \cdot 0.2 \cdot 0.2 \cdot 0.8) + 0.4 \cdot (0.5 \cdot 0.3 \cdot 0.2 \cdot 0.2 \cdot 0.5 \cdot 0.8 \cdot 0.8 \cdot 0.2)} \approx 0.07$

(d) *Other actors performances were amazing and it's clear that a lot of effort, passion and planning went into this movie.*

The words to analyse in this case are "actors","performances","amazing","effort", "passion","planning". Therefore, the final result is.

$P(S = +1|W_1 = \text{actors}, W_2 = \text{performances}, W_3 = \text{amazing}, W_4 = \text{effort}, W_5 = \text{passion},$
$W_6 = \text{planning}$
$= \frac{0.6 \cdot (0.8 \cdot 0.8 \cdot 0.9 \cdot 0.8 \cdot 0.9 \cdot 0.8)}{0.6 \cdot (0.8 \cdot 0.8 \cdot 0.9 \cdot 0.8 \cdot 0.9 \cdot 0.8) + 0.4 \cdot (0.2 \cdot 0.2 \cdot 0.1 \cdot 0.2 \cdot 0.1 \cdot 0.2)} \approx 0.999968$

$P(S = -1|W_1 = \text{actors}, W_2 = \text{performances}, W_3 = \text{amazing}, W_4 = \text{effort}, W_5 = \text{passion},$
$W_6 = \text{planning}$
$= \frac{0.4 \cdot (0.2 \cdot 0.2 \cdot 0.1 \cdot 0.2 \cdot 0.1 \cdot 0.2)}{0.6 \cdot (0.8 \cdot 0.8 \cdot 0.9 \cdot 0.8 \cdot 0.9 \cdot 0.8) + 0.4 \cdot (0.2 \cdot 0.2 \cdot 0.1 \cdot 0.2 \cdot 0.1 \cdot 0.2)} \approx 0.000032$

3 Continuous distribution + Exponential

Let X be a random variable following an Exponential distribution with rate parameter $\lambda > 0$, representing the time until the occurrence of an event. The memoryless property of the exponential distribution is proven by demonstrating that, for any non-negative constants a and b, the following holds:

$$P(X > a + b|X > a) = P(X > b)$$

Demonstration of the property: First of all, it is needed to start with the definition of conditional probability,

$$P(X > a + b|X > a) = \frac{P(X > a + b \wedge X > a)}{P(X > a)}$$

Since $X > a + b$ implies $X > a$, it is possible to simplifies the numerator to $P(X > a + b)$. Therefore,

$$P(X > a + b|X > a) = \frac{P(X > a + b)}{P(X > a)}$$

Now, let's use the probability density function (pdf) of the exponential distribution with the rate parameter λ is defined as:

$$f(x; \lambda) = \lambda e^{-\lambda x}$$

Then the probabilities can be written in terms of the integrals applied to the pdf of exponential distribution :

$$P(X > a + b|X > a) = \frac{\int_{a+b}^{\infty} \lambda e^{-\lambda x}\, dx}{\int_{a}^{\infty} \lambda e^{-\lambda x}\, dx}$$

Computing the integrals,I obtain that:

$$P(X > a + b|X > a) = \frac{e^{-\lambda(a+b)}}{e^{-\lambda a}}$$

Simplifying, the expression is:

$$P(X > a + b|X > a) = e^{-\lambda b}$$

Note that :

$$P(X > b) = \int_{b}^{\infty} \lambda e^{-\lambda x}\, dx = e^{-\lambda b}$$

Hence,

$$P(X > a + b|X > a) = P(X > b)$$

I can conclude that the two expressions are equal and the memoryless property of the exponential distribution is proved. This conclusion means that this unique characteristic of that distribution implies that the probability to wait an additional time unit b, despite a time units have already passed, is equal to the probability to wait b time units without considering any already passed waiting time. In other words,the probability of having already waited a time units does not influence the probability of an event to occur in a future moment.Basically, the distribution of remaining time until the event does not depend on the past.

4 Linear regression + estimation

The aim of a study is to find out the effect of exposure to sunlight on the increase of freckles per exposure hour. A study of 100 people find that the effect is $\hat{\beta} = 0.3$.

a) Interpret the coefficient.
The coefficient β , in this context , represents the estimate value of increasing the freckles each hour under sunlight exposure by an average of 0.3.

b) Then a bootstrap is performed. Explain in detail.

   i. How the bootstrap works.

   The bootstrap is a statistical method used to estimate the sampling distribution of a statistic (like mean, variance, median and so on) by generating multiple samples (with replacement) from the original dataset. It is possible to identify 3 main steps for computing a bootstrap:

   1. Create a sample with replacement by getting n observations from the original sample of size n
   2. Calculate the selected statistic
   3. Repeat the previous both phases in order to create a distribution of statistics

ii. The purpose of the bootstrap.

The purpose of the bootstrap method is to estimate the sampling distribution of a statistic (like the mean, variance, model coefficients and so on) when the true distribution is unknown or difficult to calculate directly. Indeed, through multiple resamplings of the original data it is possible to generate many simulated samples that allows to understand better the behaviour of the statistic and the nature behind the original data ,for example if it can be useful to see if the data belong to the same distribution of information or not by analyzing the samplings distribution .

5 Logistic regression + prediction

a) Explore the role of `logit` and `logistic` function in the logistic regression. Find one function as the inverse of the other.

In the logistic regression the `logit` function :

$$logit(p) = \ln(\frac{p}{1-p})$$

where p is the probability of the outcome.
The `logit` function is use to convert a probability to its corresponding real number value and this concept allows to model a linear combination of the predictor variables in the form:

$$logit(\pi_i) = X_i\beta$$

where π_i is the dependent variable assumed as a Bernulli distribution which represents the likelihood of success for the observation i. I can return whichever real numbers between $(-\inf, +\inf)$.

Talking about the `logistic` function ,also named "sigmoid" function, is the function :

$$\pi_i = \text{logistic}(X_i\beta) = \frac{1}{1+e^z}$$

where z is the linear combination of predictor variables :

$$z = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \cdots + \beta_n x_n$$

where β represents the model coefficients and X the predictive variables.
This function returns a probability value given predictor variables in order that the dipendent value is equal to 1 $\pi_i = P(Y = 1)$ where Y represents the dipendent variable.Thus, it returns values between 0 and 1.

To summarize the concepts. in logistic regression, the logit function is used to express the logarithm of the probability as a linear relationship with the predictor variables and,instead, the logistic function is used to convert these linear model back into a probability.

It is also possible to notice that `logit` function represents the inverse function of `logistic` indeed it can be proved by a simple proof:

$$f(x) = \text{logit}(x)$$

$$y = \ln(\frac{x}{1-x})$$

Then I remove the logarithm:

$$e^y = \frac{x}{1-x}$$

$$
\begin{aligned}
(1-x)e^y &= x \\
e^y - xe^y &= x \\
x(e^y+1) &= e^y \\
x &= \frac{e^y}{1+e^y} \\
x &= \frac{1}{1+e^{-y}}
\end{aligned}
$$

Therefore, it can assume that:

$$f^{-1}(x) = \text{logistic}(x) = \frac{1}{1+e^{-x}}$$

b) How do you evaluate the accuracy? Would you evaluate it in your training or test data? Explain why
To evaluate the accuracy of a logistic regression model there are many metrics to analyze how the model behaves such as :

(a) Accuracy: The ratio of correct predictions on the total number of predictions.

(b) Precision: The ratio of true positives predictors on total number of true positive and false positive predictions.

(c) Recall : The ratio of true positives predictors on total number of true positive and false negative predictions.

(d) F1 Score : The harmonic mean of precision and recall.

(e) Kappa : Statistic that measures inter-rater agreement for categorical items, adjusting for change agreement

Despite all those metrics , it is necessary to use test data instead of training ones in order to correctly analyze the model behaviour because of the fact that test data allow to show how the model can perform with new and unseen data.

6 Hypothesis testing

In a recent music festival, organizers measured the duration of unplanned pauses during 100 performances due to suspected technical problems. The mean duration was found to be 1.7 seconds.

(a) State the null and the alternative hypothesis.
The null hypothesis affirm that *the pauses duration is 1.7 seconds*, it can be also written as $H_0 = 1.7s$ and the alternative hypothesis confirms that *the pauses duration is not 1.7 seconds* also showed as $H_1 \neq 1,7s$

(b) The subsequent analysis yielded a p-value of 0.0001. Did the organizers guess correctly?

Yes, the organizers guess correctly to suspect about the duration of pauses .This final closure is based on the fact that the p-value assumes a small number and when this value is less then a specific threshold (like 0.01 or 0.05 generally) it means that the null hypothesis has to be rejected H_0 in order to affirm the alternative hypothesis H_1 . Therefore , the organizers' suspect about the technical problems is statistically supported by the strong evidence against the null hypothesis abetting the alternative one.

7 You are the CEO of a start-up company

Just imagine that you are a CEO of a start-up company in medical diagnostics. You are planning to launch a new diagnostic tool to detect diabetes in women using easily available information without invasive testing. You want to convince investors to invest in your company and you need to given them a report with the potential of your test.

You want to combine easily available values, such as

(a) **pregnant**: number of times pregnant

(b) **glucose**: Plasma glucose concentration at 2 hours in an oral glucose tolerance test

(c) **diastolic**: Diastolic blood pressure (mm Hg)

(d) **triceps**: Triceps skin fold thickness (mm)

(e) **insulin**: 2-Hour serum insulin (mu U/ml)

(f) **bmi**: Body mass index (weight in kg/(height in metres squared))

(g) **age**: Age (years)

with a new test developed by your company, called **diabetes**, which measure a score based on the prevalence of diabetes in your family.

You have performed a clinical trial with 768 women, where you also recorded whether or not the women had diabetes (the outcome of test). The data is available on the iCorsi website. Write a report covering the following points

(a) **Research problem**. Formulate a research question (Hint: it should contain two elements).

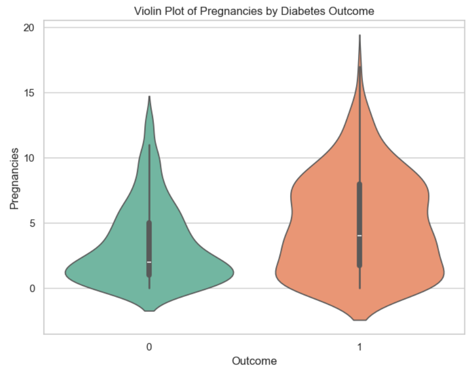
How effectively can the combination of non-invasive indicators such as the number of pregnancies, plasma glucose concentration, diastolic blood pressure, triceps skin fold thickness, serum insulin levels, body mass index (BMI), and age, along with a familial predisposition score, predict the presence of diabetes in women, compared to standard diagnostic methods that use invasive tests?

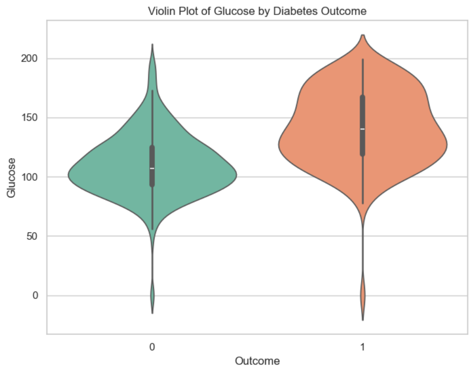
(b) **Exploratory analysis**. Perform an informal analysis of the data (i.e. tables and/or plots of the data) relevant for answering the research question.

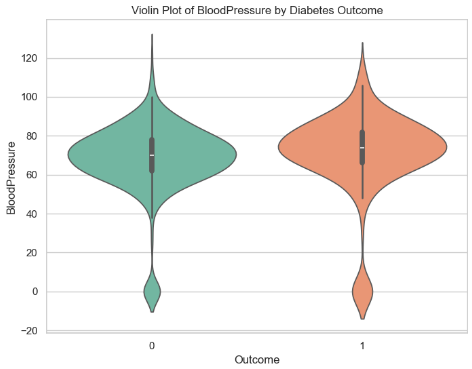
i. Evaluate the distribution of each variable, including **pregnant**, **glucose**, **diastolic**, **triceps**, **insulin**, **bmi**, **age**, and **diabetes**. Look for patterns, outliers, or anomalies using histograms, box plots, or other graphical methods.

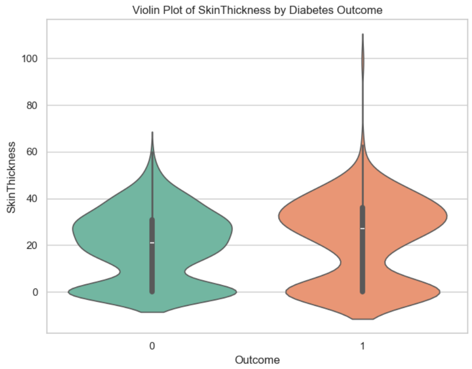
First of all , I create two violin plots for each variable that show the data distribution. I decide to create two separate violin graphs ,one for the value distribution in which the diabetes is not found and the other one for the inverse situation, because I would like to make more comprehensible the data readability instead of having only one plot that aggregate all data , this representation could make difficult to analyze and ,eventually, find outliers, patterns and other relevant information about structure of data compared to my choice.

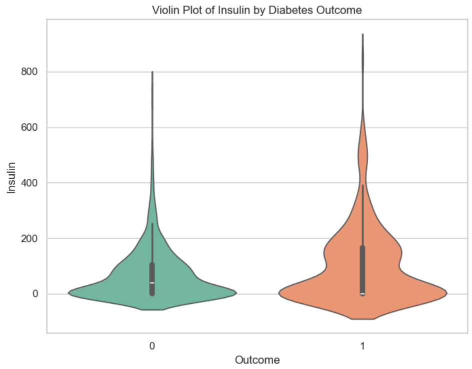
The violin graphs I obtain are the following:

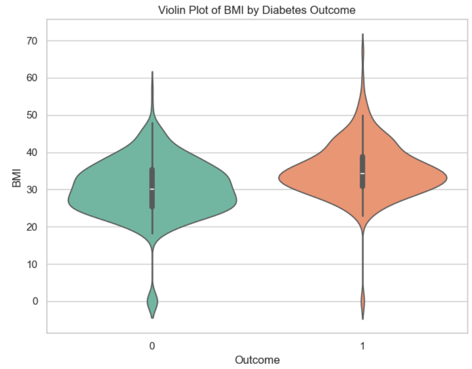

Violin Plot of Pregnancies by Diabetes Outcome

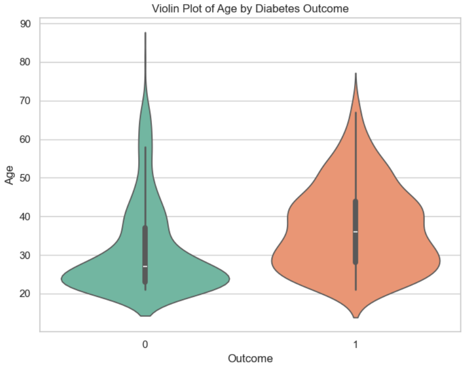

Violin Plot of Glucose by Diabetes Outcome


Violin Plot of BloodPressure by Diabetes Outcome


Violin Plot of SkinThickness by Diabetes Outcome


Violin Plot of Insulin by Diabetes Outcome


Violin Plot of BMI by Diabetes Outcome



Note : the code for generating the previous plot is available in file named `violinPlots_generator.py`

Before beginning the analysis of each variable, it is important to note that an informal examination of the CSV data file (which is also evident in the graphs) shows that several fields for various information have a value of zero. This likely indicates that the data in question were not recorded, perhaps because they would not be reliable (for example, glucose values cannot realistically be zero and are more likely to be unrecorded). Therefore, these "anomalies" must be considered given that they can impact the structure of the distribution, especially if the number of zero entries is significant.

In particular, in some distributions, such as the triceps and insulin ones, the anomalies due to zero values are substantial and very apparent, creating a skewed distribution.

Furthermore, it is possible to observe from the various graphs that the distributions take on more or less well-defined models: For example,the glucose, diastolic, and BMI graphs, their models are approximately close to a normal distribution (both when diabetes is recorded and when it is not), despite the anomalous values due to zeros in the distribution. While the distributions of age, triceps, insulin, and pregnancies are quite unbalanced in terms of distribution:

Indeed, it is possible to note that the graphs for insulin and skin thickness are characterized by a wide distribution of zeros and, therefore, measurements not taken for both left and right plots.Since, for age and pregnancies, the plots are more concentrated on lower values (indicating a higher number of younger women and a prevalence of women who have had few childbirths) in individuals with no diabetes condition (left plots). On the other hand,in the diabetes recorded cases (right graphs), both distribution appear to almost follow a normal distribution.

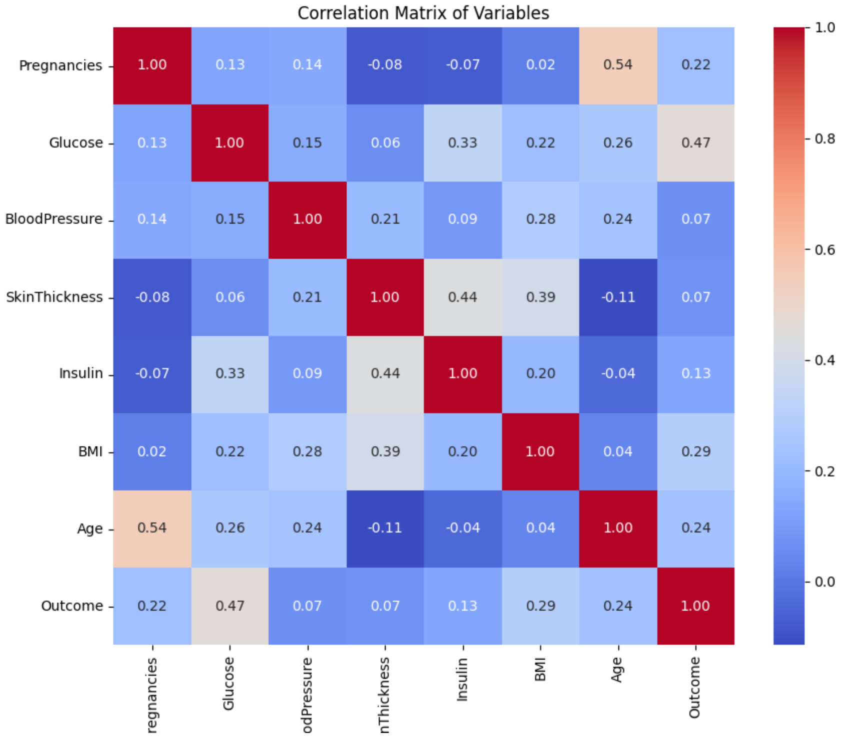
ii. Check the completeness of the data. Search for missing or blank values in all the variables and consider their potential impact on your analysis.

As mentioned, the presence of empty or zero-marked values complicates an accurate data analysis due to these anomalies. Indeed those values can affect the predictive model's performance. Decisions on how to handle these missing values can be crucial for obtaining a better result.

iii. Investigate the interactions between variables. Use scatter plots or correlation matrices to identify relationships or trends, particularly those that might be crucial in predicting the outcome of the **test**.

To analyze the interactions between variables, I decide to use a correlation matrix to show which is the grade of correlation between the elements. As it is well-known, if the value of the correlation is near 1 , it means there is a positive iteration between variables,thus both variables increase if one of them increment its value, but in the case in which the value is near -1 , there is the negative opposite situation and variables have a inversely related behaviour.

The correlation matrix I obtained is the following:



From the matrix, it is possible to notice that, in addition to various auxiliary information, such as the positive relationship between age and pregnancies, the row referring to the Outcome variable relationship with other variables dictates the correlation values which define how the various data points impact the presence of diabetes. For example, the variables which appear to have a greater likelihood of influencing the presence of diabetes in a woman are glucose (0.47), BMI (0.29), age (0.24), and pregnancies (0.22), while the other variables have a lower influence on this result. It should be remembered that these values do not necessarily indicate the presence of diabetes; however, the correlation matrix has enabled the identification of which variable values might have a more significant impact on predicting the test results.

(c) **Formal analysis**. Perform a formal analysis of the data based upon several predictors.

i. Split the data into a training set (90%) and a test set (10%) using your student number as seed of the random number generator. Use your own function to split the data.

Once I create the Linear Congruential Generator `lcg` algorithm function to generate pseudo-random numbers for the indexes , I aim to generate the 10% of data indexes contained in the file for obtaining the test set and subsequently creating the training set from the remaining percentage of data. Next I inserted the code I used to generate the data indexes that have to be divided in the two types of sets:

```
seed=1917953388 #seed
np.random.seed(seed) #used for reproducibility in bootstrap analysis

#Linear Congruential Generator (LCG) function
def lcg(seed, a, c, m, n, rescale):
    x = seed
    randNums= []
    for i in range(0,n):
        x = (a * x + c) % m
        if(rescale):
            x /= m
        randNums.append(x)
    return randNums

#File name
file_path = ’diabetes.csv’

# Load the ’diabetes.csv’ dataset
data = pd.read_csv(file_path)

number_of_rows = len(data) #768
test_set_size = int(0.1*number_of_rows) #76

#Parameters for the LCG
a=3
c=1
m=number_of_rows

#Calculate the indices for the test set (10% of the data)
test_indexes = lcg(seed,a,c,number_of_rows,number_of_rows,False)

test_indices = list(set(test_indexes))  # Ensure uniqueness of indexes

#control if the test set size is correct
if len(test_indices) > test_set_size:
    test_indices = test_indices[:test_set_size]  # Trim to the desired size


# Split the data into two arrays based on indices (training and test sets)
test_data = data.iloc[test_indices] #test set : 10% of the data
training_data = data.drop(test_indices) #training set : 90% of the data
```

ii. Fit the test score as a function of the other variables using your own function. Use as input of this function the response and the predictor variables of the training set and as output the estimate of the coefficients.

I wrote the following Python code in order to compute the coefficients to be subsequently applied to the test dataset in order to check the accuracy of the predictive model:

```
# Prepare the data for logistic regression
# Extract features and target variables for training and testing sets
X_train = training_data.drop(’Outcome’, axis=1).values
y_train = training_data[’Outcome’].values

X_test = test_data.drop(’Outcome’, axis=1).values
y_test = test_data[’Outcome’].values

# Adding intercept term to both X_train and X_test (beta_0)
n_train = X_train.shape[0]
X_train = np.c_[np.ones(n_train), X_train]

n_test = X_test.shape[0]
X_test = np.c_[np.ones(n_test), X_test]

#Function definitions for the logistic regression

# Logistic function
def sigmoid(x):
    return 1 / (1 + np.exp(-x))

# Predict function
def predict(X, beta):
    return sigmoid(np.dot(X, beta))

#Gradient of the loss function
def log_loss_grad(y, X, beta):
    predictions = predict(X, beta)
    return np.dot(X.T,predictions - y) / len(y)

# Gradient Descent
def gradient_descent(X, y, alpha, epsilon, max_iter):
    columns=X.shape[1];
    beta = np.zeros(columns)

    i = 0
    difference = 99999.9

    while i < max_iter and difference > epsilon:
        grad = log_loss_grad(y, X, beta)
        new_beta = beta - alpha * grad
        difference = np.linalg.norm(new_beta - beta)

        beta = new_beta
        i += 1

    return beta

# Parameters
alpha = 0.01   # Learning rate
epsilon = 1e-4 # Convergence threshold
max_iter = 10000 # Maximum number of iterations

beta = gradient_descent(X_train, y_train, alpha, epsilon, max_iter)
```

The outputs related to the beta coefficients and the test model accuracy applied on test dataset I obtain are the following:

```
Beta coefficients: [-3.53546025  4.84139274  0.38230806 -1.12058411
0.1427576  -0.20224153 -0.28690625 -0.47279323]
Accuracy on the test set: 0.6710526315789473
```

Note : more information about the code are available in the file `diagnosticTool.py`

Where the first coefficient refers to the intercept term β_0 of the model and the other ones are the respective betas of each variable that I would like to analyze for the predictive model. Notice that the algorithm does not converge before the maximum number of iterations despite I use a small threshold value $epsilon = 1e - 4$, probably this behaviour is due to the low value of $\alpha = 0.01$ that does not help to reach fastly the convergence of algorithm.

iii. Interpret the coefficients associated with the variables **pregnant** and **diastolic**.

The coefficients,that I achieved after the logistic regression model computation, represents how each variable impact on the model $z = \beta_0 + \beta_1 x_1 + \cdots + \beta_n x_n$ and , as a consequence, on the final probability calculated by the `logistic` function.

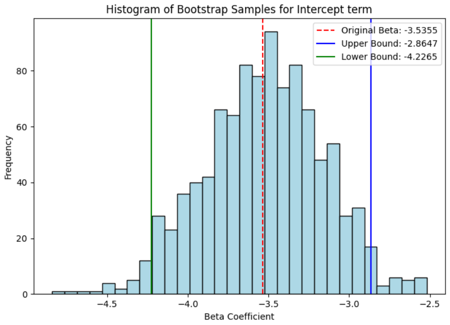
In particular I would like to focus the value assumed by the coefficient related to variables `pregnant` and `diastolic`.

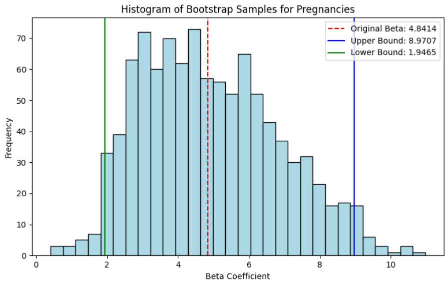
The former coefficient value I obtain is equal to $\beta_{pregnant} = 4.84139274$, this value represent a quite strong impact on the model indeed if the number of pregnancies increases , the probability to have the diabetes condition raises. Moreover, this coefficient hints that the number of pregnancies has a certain impact on the risk to have the diabetes. On the other hand, the latter coefficient number $\beta_{diastolic} = -1.12058411$ represents a negative relation with the blood pressure trend. Indeed this value can imply that higher diastolic blood pressure values has a minor risk to develop the diabetes health condition . Naturally those coefficients only represent the relationships between variables analyzed on the actual data, indeed, the value of each of them do not necessary imply the main cause of health condition.

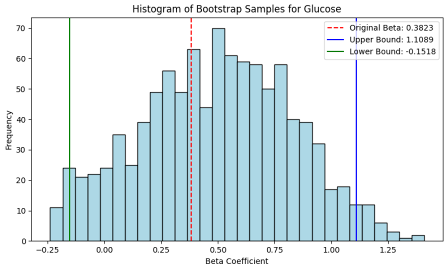
iv. Perform bootstrap (with B=1000 bootstrap samples) and test whether the coefficients from (ii) are significant at the $\alpha = 5\%$ significance level.

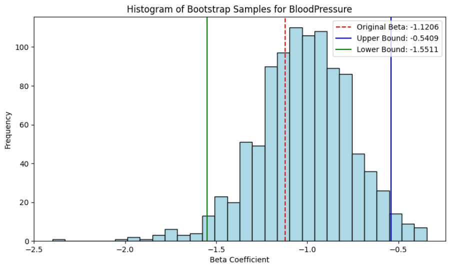
Once I perform the bootstrapping on my coefficients, I obtain the following plots in which I added a red line to show the value of the original beta coefficient , a green and a blue ones used to show respectively the lower and upper bound of the confidence interval at 95% where over those limits is represented the significance level $\alpha = 5\%$.

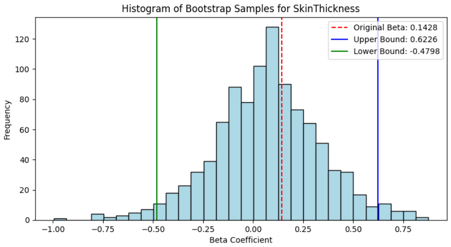
The plots I achieve are the following:

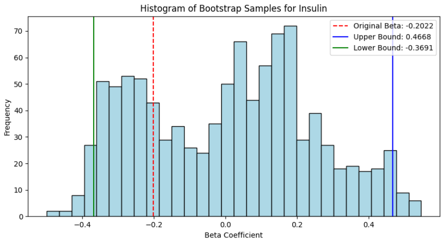


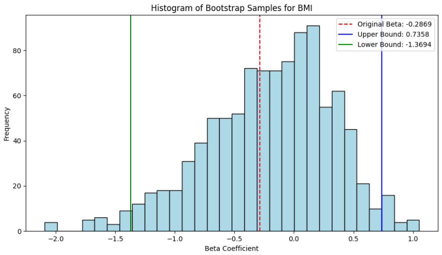


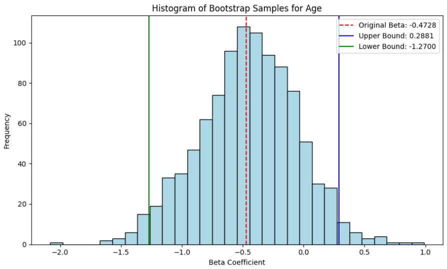












Note: the code for computing the bootstrapping is available in the file `diagnosticTool.py` and the execution of this code section can take some time to show the results (almost 3 minutes)

Now, looking the plots, it is necessary to find out which are the significance variables for the logistic regression model. A variable it is said to be significance if the value 0 is not contained inside the range limited by the confidence interval defined on the bootstrap distribution. Indeed ,when a bootstrap analysis is performed to estimate how the variable can change based on multiple data resampling with replacement, the confidence interval of the distribution is set with a given percentage in order to analyze how much impact has the variable at issue on the response variable (that in our case it is developing the diabetes health condition). This interval aims to determine the statistically significant of the variable looking for whether it includes the value 0 or not.In this case where the significance level is set to $\alpha = 5\%$, If the zero is not contained inside the 95% of the confidence range,it can be concluded that the effect of the variable in question is statistically significant for the model,otherwise, it means that there is less than a 5% probability that the true effect is null, purely by chance.

v. Select the model with only significant variables as the proposed diagnostic test. Fit the new model on the training data and evaluate its performance on the test data in terms of false positive and false negative rates.

First of all, I decided which were the significance variables of the model based on my bootstrapping analysis and I obtain that meaningful coefficients of the model are the ones related to variables of **intercept term**, **pregnancies** and **distolic**. Then, I compute a new model with only those variables using the related training data.

```
# Find significance variables indexes
significant_indices = [variables.index(variable) for variable in
                                            significant_variable]

#Save the significant variables dataset in an array both for training and for
                                             test sets
X_train_significant = X_train[:, significant_indices] # training set
X_test_significant = X_test[:, significant_indices] # test set

#Readapting the model to the significant variables
beta_significant = gradient_descent(X_train_significant, y_train, alpha,
                                            epsilon, max_iter)
```

The beta coefficients of the new model respectively of each variable are the following:

```
Beta coefficients (new model) : [-2.2286741   1.2196259   0.05977928]
```

After that, I evaluate the performance grade of the model coefficients with the test dataset computing false positive and false negative rates of the new model.

```
# Predict on the test set
y_pred_significant = predict(X_test_significant, beta_significant) >= 0.5

# Evaluating the model
true_positive = np.sum((y_test == 1) & (y_pred_significant == 1))
true_negative = np.sum((y_test == 0) & (y_pred_significant == 0))
false_positive= np.sum((y_test == 0) & (y_pred_significant == 1))
false_negative= np.sum((y_test == 1) & (y_pred_significant == 0))
```

These are the final results I achieve from the code computation:

```
False Positive Rate (FPR): 0.9622641509433962
False Negative Rate (FNR): 0.0
```

Note : extra information about the code are available in the file `diagnosticTool.py`

It is possible to notice that the model is prone to incorrectly classify negative instances as positive (FPR$\approx$ 0.96 is quite high). I believe that this tendency is due to the direct influence of significance variables passed to the model,indeed, as a consequence of their meaningful impacts, the model is more likely to predict positive outcomes (even when they should not be positive).Therefore, this model can not be a valid model for real diagnostic tests given that it could diagnose incorrectly the diabetes condition in a person at 96%.
On the other hand, the FNR= 0.0 could mean that the model is really efficient to identify true positive cases and it does not miss any , and for diabetes prediction analysis is a perfect result because it guarantees that no certain diabetes cases is avoided. Probably these two metrics results can depend on different reasons: for example there could be an overfitting of training data or the high presence of zero values inside the training dataset may be possible explanations that lead to obtain skewed FPR and FNR values.

(d) **Conclusions**. Answer the initial research problem.

After all the previous analysis it can be concluded that using easily available information such as the number of pregnancies, plasma glucose concentration, diastolic blood pressure, triceps skin fold thickness, serum insulin levels, body mass index (BMI), and age, knowing also the familial predisposition score, it is possible to predict exactly the presence of diabetes in women. Although this kind of analysis is feasible, there are some precautions to do about the diagnostic procedure to estimate the diabetes health condition. For example, as it is possible to see from analysis, the percentage of accuracy for correctly identify the diabetes developing ,despite it is quite high ($\approx$ 0.67), using this kind of diagnosis is not the best and precise way to check for the health condition compared to the use of standard invasive tests and it can incur in incorrect evaluations but a the same time this diagnostic tool offers a promising and patient-friendly approach for early detection and ongoing management of diabetes risk.

(e) **Discussion**. Provide a brief discussion about some critical aspects in the analysis

Talking about the analysis, I encountered several aspects that impact on the final conclusions of my study during the exploratory and formal phases: one of those key points was undoubtedly the presence of zero values inside the dataset. Indeed, those outliers are information that altered the evaluation of the data and had a negative effect on both the data distribution analysis and the results of the logistic prediction model.

In the former case , due to the high presence of anomalous values, some distributions appear to be strongly influenced by those values (as it happens with the insulin graph) in the exploratory analysis; in the latter situation,instead, their impact can be notice on the percentage of accuracy of the model. It is a matter of fact that the percentage of estimation ≈ 0.67 was also influence by the presence of outliers and those ones have influenced in some ways the final estimation rate. Probably if the data were cleaned from these unnecessary information and were used valid data, the diagnostic tool could have been even more accurate in searching for possible diabetes risks.

8 Bonus

Make a video to accompany your report

The video I made is a brief introduction to what I did in my analysis, the file is named `bonusVideo.mp4`
Note : the quality of the video is very low due to the limitation on file size of the iCorsi platform