Correlation and Regression

Minerva University

CS51: Formal Analyses

Prof. Volkan

Jan. 28, 2023

Introduction

The global prevalence of obesity has skyrocketed (Oliveira, G., 2015), and studies demonstrate the need for aboriginal-designed initiatives for obesity management (Sherriff, S., 2019). This paper explores the Pima Indians Diabetes Database to determine if there is a correlation between tricep skinfold thickness and Body Mass Index (BMI) as a cost-effective and accessible obesity detection method. A previous paper found both practical and statistical significance in support of the claim that individuals with a high BMI have higher skinfold thickness ($p < 0.05, d = 1.3680$). This research will continue that line of research by performing Ordinary Least Squares regression and significance-testing to analyze the data.

Dataset

The Pima Indians Diabetes Database (UCI Machine Learning, n.d.) is a subset of the data collected by the National Institute of Diabetes and Digestive Kidney Diseases, with all records belonging to female individuals of at least 21 years of age with Pima Indian heritage. Each record corresponds to a patient and contains diagnostic measurements, such as glucose concentration and blood pressure, as well as whether the patient has diabetes or not.

This report performs statistical procedures to determine the possible correlation between tricep skinfold thickness (predictor variable) and BMI (response variable). BMI is measured in $\frac{kg}{m^2}$ and is derived from a person's body height and weight. It is a quantitative continuous variable defined by the following formula:

$$BMI = \frac{w}{h^2}$$

Where w and h represents someone's body weight in kilograms and height in meters, respectively.

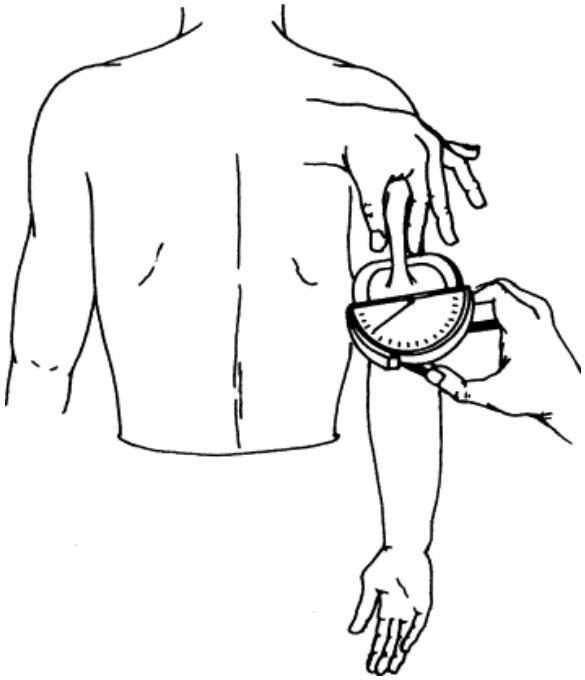


Figure 1*: An illustration of how the tricep skinfold thickness was measured in this dataset* (Eaton-Evans, 2013).

The tricep skinfold thickness is a measurement in millimeters used to quantify the double fold of skin and subcutaneous fat (Eaton-Evans, 2013); this dataset in particular measures skinfold thickness in the triceps (TSFT) (see *Figure 1*). Per the nature of the measurement, it corresponds to a quantitative continuous variable, however, in most cases this is rounded to the nearest millimeter which essentially turns it into a quantitative discrete variable[1].

Considering previous results, this paper approaches the question of whether or not there's a direct linear correlation between tricep skinfold thickness and BMI in the population of individuals with Pima Indian heritage. Assuming that the underlying population follows a linear relationship $y = \beta_0 + \beta_1 x$, the research question can be symbolically stated as follows:

[1] **#variables**: The application of this HC accurately identifies and classifies the response (dependent) variable (BMI) and the predictor (independent) variable (tricep skinfold thickness). Provides a description and illustrates how the variables are obtained or computed; in particular, shows the nuance produced when measuring a quantitative continuous variable (skin fold thickness) and rounding it, effectively turning it into a quantitative discrete variable.

$$H_0: \beta_1 = 0$$

$$H_A: \beta_1 > 0$$

Where y represents BMI and x represents tricep skinfold thickness.

For the analysis performed in this study, the dataset was pre-processed by removing all the missing measurements in the studied variables. See *Appendix A* for the specific procedure used.

Methods

Descriptive statistics

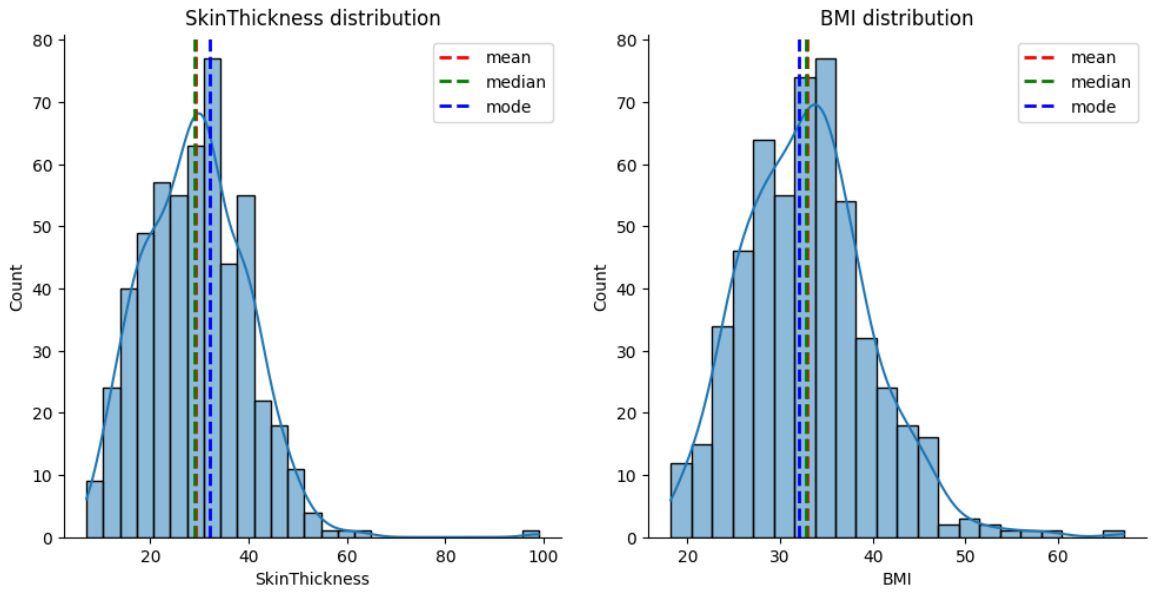


Figure 2*: Frequency distribution for tricep skinfold thickness (SkinThickness) and BMI. Each histogram displays the mean (red dashed line), median (green dashed line), and mode (blue dashed line). Both distributions show slight skewness to the right; however, the effect of the previous can be neglected, and the distributions considered quasi-normal. See Appendix C for the procedure used to create the plots.*

The pre-processed dataset has a total of 532 observations, where each entry corresponds to a unique patient's measurements. Some descriptive statistics are listed in *Table 1*, with a mean

tricep skinfold thickness of $29.18mm$ and BMI of $32.89\frac{kg}{m^2}$; see *Figure 2* for the distribution of the variables.

	SkinThickness	**BMI**
mean	29.18	32.89
std	10.52	6.88
mode	32.00	32.00
median	29.00	32.80
min	7.00	18.20
max	99.00	67.10
range	92.00	48.90

Table 1*: Descriptive statistics for the entire dataset ($n = 532$). See Appendix B for the procedure used to compute the values.*

Following the results from the previous study, we have that the, from the dataset, the individuals with high BMI ($BMI \geq 30$) have, on average, higher tricep skinfold thickness than the individuals with lower BMI ($BMI < 30$); see *Figure 3*.

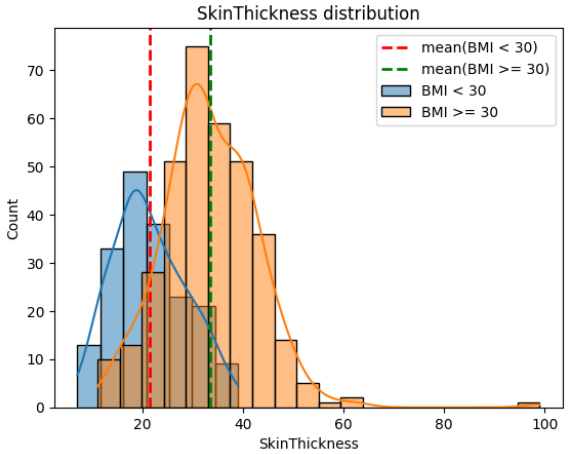


Figure 3*: Frequency distribution of tricep skinfold thickness, divided into 2 subgroups, $BMI < 30$ and $BMI \geq 30$. The red dashed line shows the mean for the first subgroup, while the green dashed line*

indicates the mean for the second. Note how the distribution and mean for $BMI \geq 30$ *is shifted to the right. Adapted from a previous assignment.*[23]

Correlation

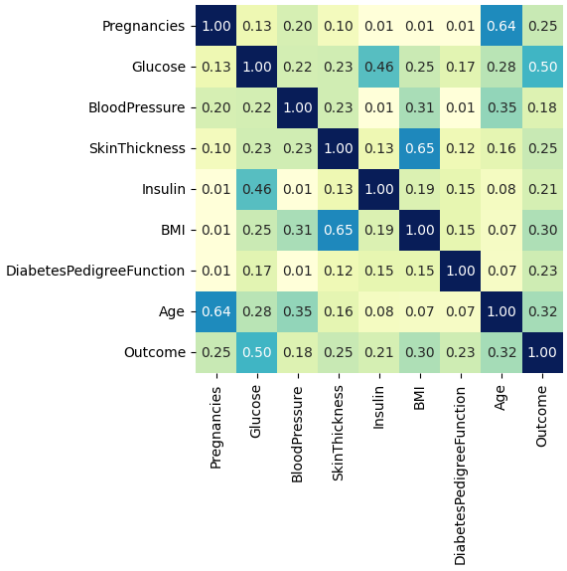


Figure 4*: A color-mapped correlation matrix showcasing the absolute value of the correlation coefficient (r) for every possible bivariate combination in the dataset. See Appendix D for the procedure used to create the plot.*

Pearson's correlation coefficient (r) and coefficient of determination (R^2) measure linear correlation between two variables. r ranges from -1 to 1: 1 for a perfect positive linear relationship, -1 for a perfect negative linear relationship, and 0 for no linear relationship; see

[2] **#dataviz**: The application of this HC provides a detailed and effective visualization of the data. It leverages previously derived results and their corresponding visualizations to explain the motivation behind the analysis decisions in this report; in particular, it takes the conclusion showcased in *Figure 3* to justify the posterior decision for a 1-tailed p-value test. Complements the descriptive statistics computed previously by showcasing the means using vertical lines. Resembles the theoretical distributions studied in classes.

[3] **#descriptivestats**: The application of this HC properly computes the descriptive statistics for the entire dataset, and uses previously computed stats for relevant subgroups to justify the analyses performed in this report. The interpretations of statistics are justified and correct.

Figure 4 for the magnitudes of r for all the variables in the dataset. R^2 ranges from 0 to 1, with 1 meaning the independent variable explains all variation in the dependent variable, and 0 meaning the independent variable does not explain any variation. Both coefficients are determined by equations.

$$r = \frac{\sum_{i=1}^{N(x)} (x_i - \bar{x})(y_i - \bar{y})}{\sqrt{(\sum_{i=1}^{N(x)} (x_i - \bar{x})^2)(\sum_{i=1}^{N(x)} (y_i - \bar{y})^2)}}$$

$$R^2 = r^2$$

Where $N(x)$ is a function that returns the number of elements in the set x, x_i the i-th element of the set x, and $\bar{x}$ the arithmetic mean of the elements in the set x. *Idem* for y.

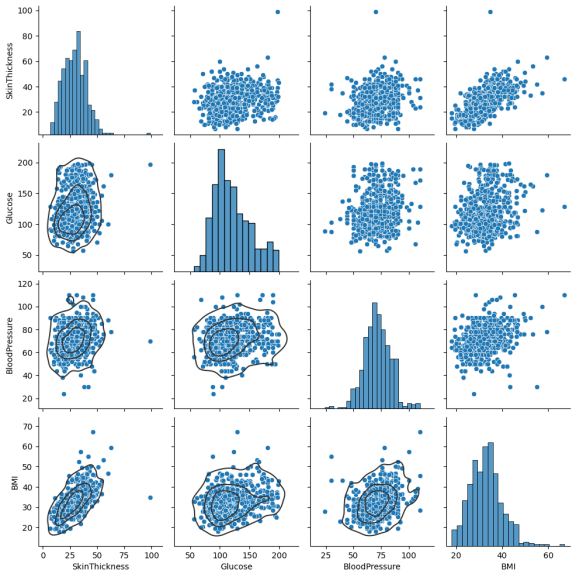


Figure 5*: A scatter matrix showcasing the relationships between arbitrarily selected variables in the dataset (SkinThickness, Glucose, BloodPressure, BMI). The main diagonal shows the frequency distribution for the corresponding variable. The lower triangle presents a normal scatter plot with an added kernel density estimation (levels = 4) to illustrate the density. See Appendix G for the procedure used to create the plots.*

Figure 4 shows the absolute value for the r metric for all the pairs of variables in the studied dataset as a correlation matrix; *Figure 5* shows pair scatter plots with kernel density estimates for arbitrarily chosen variables. Recalling previous subsections, we have that our variables of interest, tricep skinfold thickness ($SkinThickness$) and BMI have the highest correlation as measured by r.

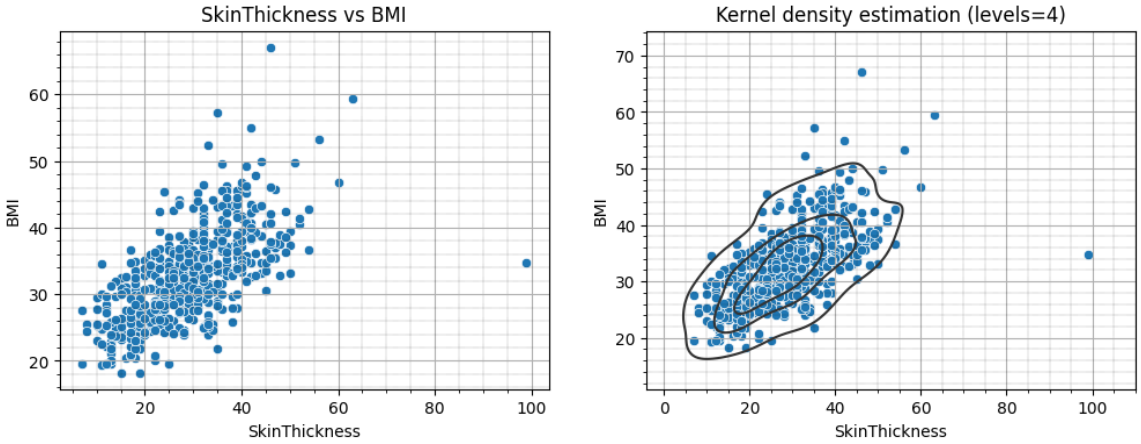


***Figure 6**: On the left: a scatter plot for tricep skinfold thickness against BMI. On the right: idem with an added layer for kernel density estimation (levels = 4) to showcase the density. Note how the kernel density estimation suggests that the outliers in the distribution do not represent a significant deviation. See Appendix E for the procedure used to create the plots.*

More specifically, r and R^2 for tricep skinfold thickness and BMI in the studied database for female individuals of Pima Indians heritage have the following values:

$$r \approx 0.6474$$

$$R^2 \approx 0.4191$$

r indicates a strong positive linear correlation; precautions must be taken, however, since *Figure 6* shows slight heteroscedasticity. This, therefore, does not imply that the increase in BMI is caused by the predictor variable. There can be confounding variables, such as body fat (Sevoyan, A., 2019), responsible for the underlying causation.

R^2 indicates that approximately 42% of the variation in BMI can be explained by the linear regression model with tricep skinfold thickness. This is an intuitively high value, which further supports the claim of a strong linear relationship[45].

Simple linear regression

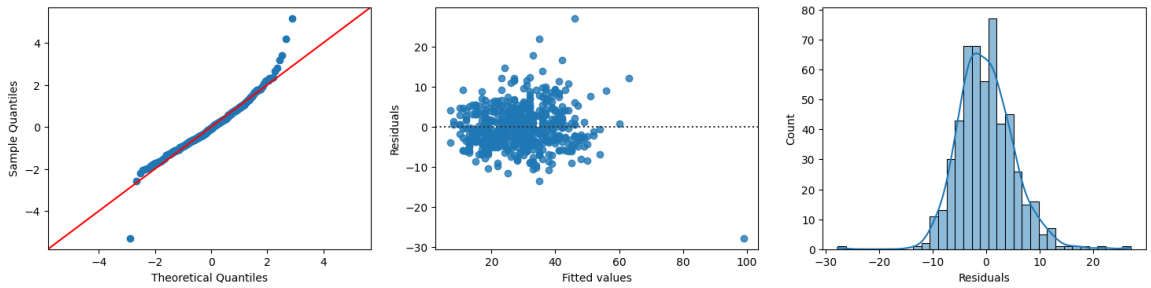


***Figure** 7: Diagnostic plots. On the left: a qq-plot for the residuals indicating that there is no significant deviation from the theoretical normal distribution. In the middle: a residual plot showcasing that the mean of the residuals is approximately 0. On the right: a frequency distribution for the residuals showcasing that the residuals do not present a significant deviation from the theoretical normal distribution.*

Ordinary Least Squares (OLS) regression is used to estimate the linear relationship between BMI and tricep skinfold thickness. It assumes independent and normally distributed errors; the latter is shown in *Figure 7*. The goal is to find the line that best fits the samples and use it to make predictions about the population's BMI.

$$\hat{y} = b_0 + b_1 x$$

[4] **#correlation**: The application of this HC showcases how the correlation between tricep skinfold thickness and BMI was computed through Pearson's r, suggesting that there's a linear relationship. Upon examining the scatter plot, the reliability of r was questioned due to heteroscedasticity, particularly for higher values for skinfold thickness. The difference between correlation and causation was explained and possible confounding variables were identified.

[5] **#regression**: This application of this HC correctly computes and interprets the meaning behind the determination coefficient.

$$b_0 = \bar{y} - b_1\bar{x}$$

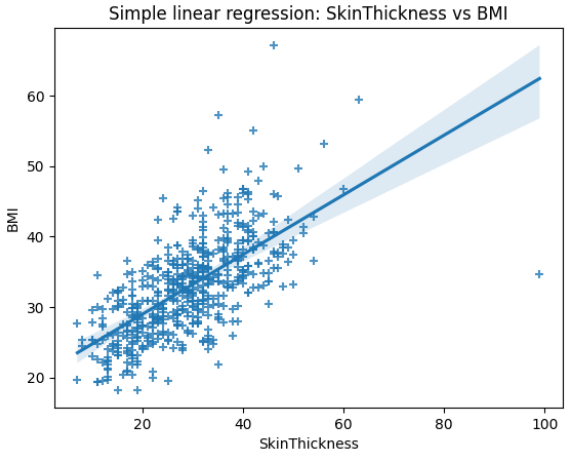
$$b_1 = r\frac{s_x}{s_y} = \frac{\sum_{i=1}^{N(x)}(x_i-\bar{x})(y_i-\bar{y})}{\sum_{i=1}^{N(x)}(x_i-\bar{x})^2}$$

Where s_x and s_y represents the sample standard deviation for the set x and y, respectively.

After the computations (see Appendix G for the procedure used for the computations and plots), the regression line (see *Figure 8*) is defined as

$$\hat{y} \approx 20.5367 + 0.4233x$$

The obtained parameters indicate that the theoretical "base-rate" for when the tricep skinfold thickness would be 0 is $b_0 \approx 20.5367$, and b_1 indicate that for every change in 1 unit of tricep skinfold thickness, in this case, millimeters, the BMI would change 0.4233 units, in this case, $\frac{kg}{m^2}$[6].



[6] **#regression**: This application of this HC showcases how a simple linear regression model (ordinary least squares) was computed, and its explanatory power, conveyed by R^2, was explained in the previous subsection. The regression coefficients were accurately interpreted.

Figure 8*: A simple regression plot for the previously computed regression between tricep skinfold thickness (SkinThickness) and BMI.*

Significance testing and confidence intervals

Significance testing and confidence intervals are tools that allow researchers to test the validity of their results. This subsection performs those statistical procedures to validate the obtained results with the hypothesis defined in previous sections.

$$y = \beta_0 + \beta_1 x$$

$$H_0: \beta_1 = 0$$

$$H_A: \beta_1 > 0$$

In particular, this section performs a 1-tailed p value test with a significance $\alpha = 0.01$, given the requirements: randomness, independence (sample less than 10% of the population), and normality (see *Figure 2*). For this purpose, it is necessary to compute the standard error for the regression line (see Appendix K for the procedure used for the computations):

$$SE_{b_1} = \frac{s_y}{s_x}\sqrt{\frac{1-R^2}{N(x)-2}} \approx 0.0216$$

Then, it is necessary to compute the t score for the obtained coefficient b_1 in terms of SE_{b_1}

$$t = \frac{b_1 - \beta_1}{SE_{b_1}} \approx 19.5567$$

Finally, the p value is computed in terms of the survival function:

$$cdf_X(x, n) = P(X \leq x)$$

$$sf_X(x, n) = 1 - cdf_X(x, n)$$

$$p = sf_X(|t|, N(x) - 2) \times 2 \approx 7.9813 \times 10^{-65}$$

Where sf is the survival function, cdf is the cumulative distribution function, and $P(X \leq x)$ is the probability function the probability that X will have a value less than or equal to x with n degrees of freedom.

Considering the initially set significance $\alpha = 0.01$, the performed significance test provides evidence for rejecting the null hypothesis H_0 ($p < 0.01$) as well as supporting the practical significance found by R^2[7].

$$ppf_X(x, n) = cdf_X^{-1}(x, n)$$

$$t = ppf_X(1 - \frac{\alpha}{2}, N(x) - 2) \approx 2.5851$$

$$l = b_1 - t \times SE_{b_1} \approx 0.3679$$

$$u = b_1 + t \times SE_{b_1} \approx 0.4792$$

$$P(\beta_1 \in [l, u]) = 95\%$$

Where $ppf_X(x, n)$ is the inverse cumulative distribution function $cdf_X^{-1}(x, n)$ or percentile function with n degrees of freedom.

This means that there's a 99% probability that the underlying population's coefficient β_1 is in the range $[0.3673, 0.4792]$. More generally this indicates that the underlying coefficient β_1

[7] **#significance**: The application of this HC computed and properly interpreted the measure of significance, obtaining statistical evidence supporting the conclusion for a linear correlation between the studied variables. All the necessary requirements were stated, in addition to the difference between practical and statistical significance.

will be contained in the 99% confidence interval for a given sample 99% of the time[8] (see Appendix L for the procedure used for the computations).

Multiple linear regression and forward selection

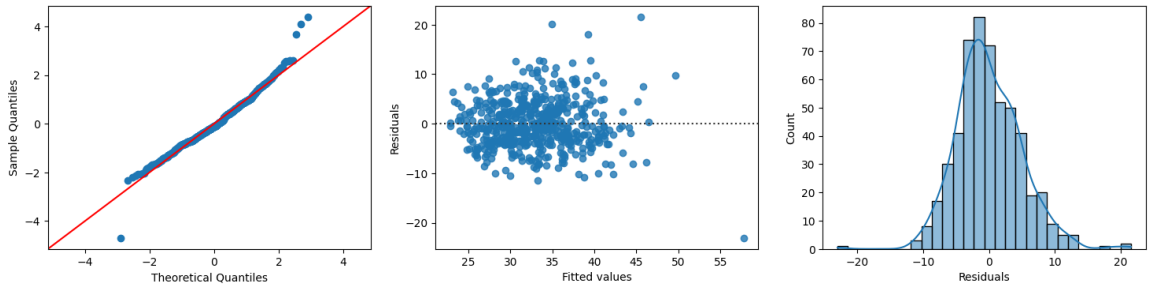


Figure 9*: Diagnostic plots. On the left: a qq-plot for the residuals indicating that there is no significant deviation from the theoretical normal distribution. In the middle: a residual plot showcasing that the mean of the residuals is approximately 0. On the right: a frequency distribution for the residuals showcasing that the residuals do not present a significant deviation from the theoretical normal distribution.*

Multiple regression is a powerful technique to study relationships between multiple predictors and a single dependent variable. Forward selection is a method of building a model by adding predictors sequentially and choosing the best model based on a selection criterion. To measure model performance, we must use $adj(R^2)$, defined as:

$$adj(R^2) = 1 - \frac{(1-R^2)(N(x)-1)}{N(x)-p-1}$$

Where p is the number of predictor variables used in the model.

To conduct forward selection, predictor variables are added sequentially and the best model is chosen based on $adj(R^2)$. The procedure is repeated until $adj(R^2)$ no longer increases,

[8] **#confidenceintervals**: The application of this HC accurately computes and interprets the mean of a 99% confidence interval. Provides complimentary statistical evidence supporting an underlying linear correlation between the studied variables.

indicating optimal fit. See Appendix I and Appendix H for the procedures used for the computations and creating the plots, and *Figure 9* for the diagnostic plots. Variables that optimized $adj(R^2)$ expressed the following equations[9] (see *Figure 10*):

$$adj(R^2) \approx 0.48$$

$$\hat{y} = 14.84 - 0.11 \times Pregnancies + 0.11 \times BloodPressure + 0.37 \times SkinThickness + 0.01 \times Insulin + 0.89 \times DiabetesPedigreeFunction - 0.07 \times Age + 1.98 \times Outcome$$

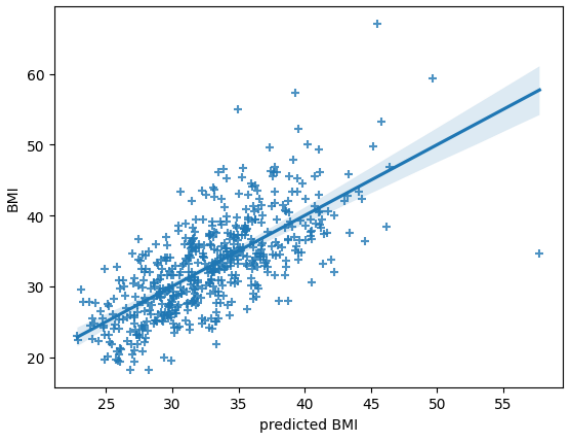


***Figure 10**: A multiple regression plot for the previously computed regression between the number of Pregnancies, BloodPressure, SkinThickness, Insulin, DiabetesPedigreeFunction, Age, Outcome, and BMI. Given the high-dimensionality of the regression, the scatter plot showcases the relationship between predicted BMI and BMI.*

Results and Conclusion

Analysis of tricep skinfold thickness and BMI in Pima Indians showed a strong positive linear relationship ($r \approx 0.6474, R^2 = 0.4191$) with 42% of the variation in BMI explained by

[9] **#regression**: This application of this HC showcases how a multiple linear regression model (ordinary least squares) was computed and its explanatory power conveyed by adj(R^2). The regression coefficients were accurately interpreted. Sets the grounds for the comparison done in the conclusion.

the variation in tricep skinfold thickness. A simple linear regression model found for every unit change in tricep skinfold thickness, BMI changed by 0.4233 units (99% CI: $[0.3673,\ 0.4792]$). A multiple regression model with forward selection identified tricep skinfold thickness, pregnancies, blood pressure, insulin, diabetes pedigree function, age, and outcome as important predictors of BMI ($adj(R^2) \approx 0.48$), which, considering the number of predictor variables, is not a significant improvement. Results suggest tricep skinfold thickness is an important predictor of BMI, but further analysis with a larger sample size is needed. Also, additional studies are required since this statistical argument only poses inductive evidence, since it draws generalizations for the population based on conclusions from a small and biased sample (contrasting with a deductive argument)[10].

1390 words[1112].

Reflection

To ensure the accuracy of my results, I used the t score formula to calculate the t score for the confidence interval and the significance test. Furthermore, I used the survival function to calculate the p value. I used the online resources from Khan Academy to review the formulas and methods used in the assignment. I also researched statistical methods used to analyze data, such as correlation and regression.

---

[10] **#induction**: The application of this HC Effectively explains why the presented evidence is an inductive argument. Furthermore, it provides the characterizing property that differentiates induction from deduction.

[11] **#professionalism**: The application of this HC follows all the conventions for the APA academic formatting and remains within the established word count. Also, the code follows Google's stylistic conventions for formatting and self-documentation.

[12] **#organization**: The application of this HC takes into consideration multiple aspects of academic communication and #scienceoflearning to present and showcase the performed procedures in a logical manner. The respective analyzes subsections present the computations in a technical language but the conclusion is intentionally written in a straightforward manner so that it's easy to understand.

I'd like to thank **Daria Khmara** for uplifting my spirit and being the best part of today (written on January 28, 2023). I'd also like to thank **Anne Behme** and **Johanna Seidel** for feeding me throughout the week and keeping me going to write this assignment.

References

Eaton-Evans, J. (2013). Nutritional Assessment: Anthropometry. *Encyclopedia of Human Nutrition (Third Edition)*, 227-232. https://doi.org/10.1016/B978-0-12-375083-9.00197-5

Oliveira, G. F., Oliveira, T. R., Ikejiri, A. T., Galvao, T. F., Silva, M. T., & Pereira, M. G. (2015). Prevalence of Obesity and Overweight in an Indigenous Population in Central Brazil: A Population-Based Cross-Sectional Study. Obesity facts, 8(5), 302–310. https://doi.org/10.1159/000441240

Sevoyan, A., Davison, B., Rumbold, A., Moore, V., & Singh, G. (2019). Examining the relationship between body mass index and adverse cardio-metabolic profiles among Australian Indigenous and non-Indigenous young adults. Scientific reports, 9(1), 3385.

Sherriff, S. L., Baur, L., Lambert, M., Dickson, M., Eades, S., & Muthayya, S. (2019). Aboriginal childhood overweight and obesity: the need for Aboriginal designed and led initiatives. Public Health Research & Practice, 29(4).

UCI Machine Learning. (n.d.). *Pima Indians Diabetes Database*. Kaggle. Retrieved December 10, 2022, from https://www.kaggle.com/datasets/uciml/pima-indians-diabetes-database

Appendix A

Python code for the dataset preprocessing procedure.

```
def preprocess_df(df: DataFrame) -> DataFrame:
    """
    Preprocesses the DataFrame by removing rows with 0 values for the
columns
    of interest.
    """

    # Remove rows with 0 values for the columns of interest.
    df = df[(df[X] != 0) & (df[Y] != 0)]
    for x in Xs:
        df = df[df[x] != 0]

    return df
```

Appendix B

Python code for the procedure to generate the summary of the descriptive statistics for the dataset.

```
def get_summary(df: DataFrame, X: str, Y: str) -> DataFrame:
    """Returns a summary of the DataFrame for the given X and Y
variables."""

    summary = df[[X, Y]].describe().T[["mean", "std", "min", "max"]]
    summary["mode"] = df[[X, Y]].mode().T
    summary["median"] = df[[X, Y]].median().T
    summary["range"] = summary["max"] - summary["min"]
    summary = summary[["mean", "std", "mode", "median", "min", "max",
"range"]]
    summary = summary.T
    return summary
```

Appendix C

Python code for the procedure to plot a histogram with a kernel distribution estimate and vertical lines for the descriptive statistics.

```
def plot_distribution(x: Series, name: Optional[str] = None):
    """Plots (and saves) a distribution plot for the given Series."""

    sns.displot(x, kde=True)

    # Plot the mean, median, and mode as vertical lines.
    plt.axvline(x.mean(), color="red", label="mean", linestyle="--",
linewidth=2)
    plt.axvline(x.median(), color="green", label="median", linestyle="--",
linewidth=2)
    plt.axvline(x.mode()[0], color="blue", label="mode", linestyle="--",
linewidth=2)

    plt.title(f"{x.name} distribution")
    plt.legend()
    plt.savefig(get_figure_filename(name or "distribution"), dpi=600)
    plt.show()
```

Appendix D

Python code for the procedure to create a correlation matrix.

```python
def plot_correlation_matrix(df: DataFrame, name: Optional[str] = None):
    """Plots (and saves) a correlation matrix for the given DataFrame."""

    sns.heatmap(
        df.corr().abs(),
        annot=True,
        cmap="YlGnBu",
        fmt=".2f", # format the numbers to 2 decimal places
        cbar=False, # remove the color bar
        square=True, # make the cells square (looks pretty :))
    )

    plt.savefig(get_figure_filename(name or "correlation-matrix"), dpi=600)
    plt.show()
```

Appendix E

Python code for the procedure to create a scatter plot with and without a kernel density estimate.

```
def plot_scatter(x: Series, y: Series, name: Optional[str] = None):
    """Plots (and saves) a scatter plot for the given Series."""

    fig, (ax1, ax2) = plt.subplots(ncols=2, figsize=(12, 4))

    ax1.grid(True)
    ax1.minorticks_on()
    ax1.grid(which="minor", linestyle=":", linewidth="0.25", color="black")
    ax1.set_title(f"{x.name} vs {y.name}")
    sns.scatterplot(x=x, y=y, ax=ax1)

    ax2.grid(True)
    ax2.minorticks_on()
    ax2.grid(which="minor", linestyle=":", linewidth="0.25", color="black")
    ax2.set_title(f"Kernel density estimation (levels=4)")
    sns.scatterplot(x=x, y=y, ax=ax2)

    # `levels` for kernel density estimation was chosen arbitrarily.
    sns.kdeplot(x=x, y=y, ax=ax2, levels=4, color=".2")

    plt.savefig(get_figure_filename(name or "scatter"), dpi=600)
    plt.show()
```

Appendix F

Python code for the procedure to create a scatter matrix with kernel density estimates on the lower triangle and histograms on the main diagonal.

```
def plot_scatter_matrix(df: DataFrame, name: Optional[str] = None):
    """Plots (and saves) a scatter matrix for the given DataFrame."""

    plot = sns.pairplot(df)
    plot.map_lower(sns.kdeplot, levels=4, color=".2")

    plt.savefig(get_figure_filename(name or "scatter-matrix"), dpi=600)
    plt.show()
```

Appendix G

Python code for the simple regression procedure with optional summaries and plots.

```
def simple_regression(
    x: Series,
    y: Series,
    plot: bool = False,
    plot_name: Optional[str] = None,
    summary: bool = False,
) -> tuple[float, float, float]:
    """Performs a simple regression on the given Series and returns the
R^2, b_0, and b_1."""

    _x = statsmodels.add_constant(x)
    model = statsmodels.OLS(y, _x)
    results = model.fit()

    # Print the summary if requested
    if summary:
        print(results.summary())

    residuals = results.resid
    r_squared = results.rsquared
    b_0, b_1 = results.params

    # Plot the results if requested
    if plot:
        fig, (ax1, ax2, ax3) = plt.subplots(ncols=3, figsize=(18, 4))

        # ax1: Make a QQ-plot for the residuals
        statsmodels.qqplot(residuals, fit=True, line="45", ax=ax1)

        # ax2: Make a residual plot
        sns.residplot(x=x, y=y, ax=ax2)
        ax2.set(
            ylabel="Residuals",
            xlabel="Fitted values",
        )

        # ax3: Make a histogram of the residuals
```

```
        sns.histplot(residuals, kde=True, ax=ax3)
        ax3.set(xlabel="Residuals")

        plt.savefig(get_figure_filename(plot_name or "simple-regression"),
dpi=600)
        plt.show()

        # Make a regression plot
        sns.regplot(
            x=x,
            y=y,
            marker="+",
        )
        plt.title(f"Simple linear regression: {x.name} vs {y.name}")

        plt.savefig(
            get_figure_filename(plot_name or "simple-regression-regplot"),
dpi=600
        )
        plt.show()

    return r_squared, b_0, b_1
```

Appendix H

Python code for the multiple regression procedure with optional summaries and plots.

```
def multiple_regression(
    x,
    y: Series,
    plot: bool = False,
    plot_name: Optional[str] = None,
    summary: bool = False,
) -> tuple[float, float, list[float]]:
    """Performs a multiple regression on the given Series and returns the
adjusted R^2, b_0, and b_1."""

    x = statsmodels.add_constant(x)
    model = statsmodels.OLS(y, x)
    results = model.fit()

    # Print the summary if requested
    if summary:
        print(results.summary())

    residuals = results.resid
    r_squared_adj = results.rsquared_adj
    b_0, *b = results.params

    y_hat = results.predict()

    # Plot the results if requested
    if plot:
        fig, (ax1, ax2, ax3) = plt.subplots(ncols=3, figsize=(18, 4))

        # ax1: Make a QQ-plot for the residuals
        statsmodels.qqplot(residuals, fit=True, line="45", ax=ax1)

        # ax2: Make a residual plot
        sns.residplot(x=y_hat, y=residuals, ax=ax2)
        ax2.set(
            ylabel="Residuals",
            xlabel="Fitted values",
        )
```

```
        # ax3: Make a histogram of the residuals
        sns.histplot(residuals, kde=True, ax=ax3)
        ax3.set(xlabel="Residuals")

        plt.savefig(get_figure_filename(plot_name or
"multiple-regression"), dpi=600)
        plt.show()

        # Make a regression plot using the predicted values and the actual
values, given
        # the high-dimensional nature of the data.
        sns.regplot(
            x=y_hat,
            y=y,
            marker="+",
        )
        plt.xlabel(f"predicted {y.name}")
        plt.show()

    return r_squared_adj, b_0, b
```

Appendix I

Python code for the forward selection procedure for multiple regression.

```
def forward_selection(
    df: DataFrame,
    xs: list[str],
    y: str,
) -> tuple[list[str], float, float, list[float]]:
    """Performs a forward selection on the given DataFrame and returns the
best R^2, b_0, and b_1."""

    best_r_squared_adj = 0
    best_xs = None
    best_b_0 = None
    best_b = None

    for i in range(1, len(xs) + 1):
        # Get all combinations of length i
        for _xs in itertools.combinations(xs, i):
            r_squared_adj, b_0, b = multiple_regression(df[list(_xs)],
df[y])

            # If the R^2 is better than the best R^2, update the best R^2
and the best xs
            if r_squared_adj > best_r_squared_adj:
                best_r_squared_adj = r_squared_adj
                best_xs = _xs
                best_b_0 = b_0
                best_b = b

    return best_xs, best_r_squared_adj, best_b_0, best_b  # type: ignore
```

Appendix J

Python code to compute a regression line's standard error.

```
def standard_error(x: Series, y: Series, r_squared: float, ddof=1) ->
float:
   """Calculates the standard error of the regression."""

   n = len(x)
   s_x = x.std(ddof=ddof)
   s_y = y.std(ddof=ddof)
   return (s_y / s_x) * np.sqrt((1 - r_squared) / (n - 2))
```

Appendix K

Python code to compute the p-value for the significance test for the linear regression.

```
def p_value(x: Series, y: Series, beta_1: float, tails: int = 2) -> float:
    """Calculates the p-value of the regression."""

    # Use the previously defined functions to calculate the p-value
    r_squared, b_0, b_1 = simple_regression(x, y)

    # Calculate the standard error of the regression
    SE_b_1 = standard_error(x, y, r_squared)

    # Calculate the t-score
    t = (b_1 - beta_1) / SE_b_1

    # Calculate the p-value using the survival function of the
t-distribution.
    # Note that the degrees of freedom is n - 2, where n is the number of
observations,
    # because we have two parameters (b_0 and b_1).
    p: float = stats.t.sf(np.abs(t), len(x) - 2) * tails  # type: ignore

    return p
```

Appendix L

Python code to compute the confidence intervals for the linear regression coefficient for the predictor variable.

```
def confidence_interval(
    x: Series, y: Series, alpha: float = 0.05
) -> tuple[float, float]:
    """Calculates the confidence interval of the regression."""

    # Use the previously defined functions to calculate the confidence
interval
    r_squared, b_0, b_1 = simple_regression(x, y)

    # Calculate the standard error of the regression
    SE_b_1 = standard_error(x, y, r_squared)

    # Calculate the t-score. Note that the degrees of freedom is n - 2,
where n is the number of observations,
    # because we have two parameters (b_0 and b_1).
    t: float = stats.t.ppf(1 - alpha / 2, len(x) - 2)  # type: ignore

    lower = b_1 - t * SE_b_1
    upper = b_1 + t * SE_b_1

    return lower, upper
```