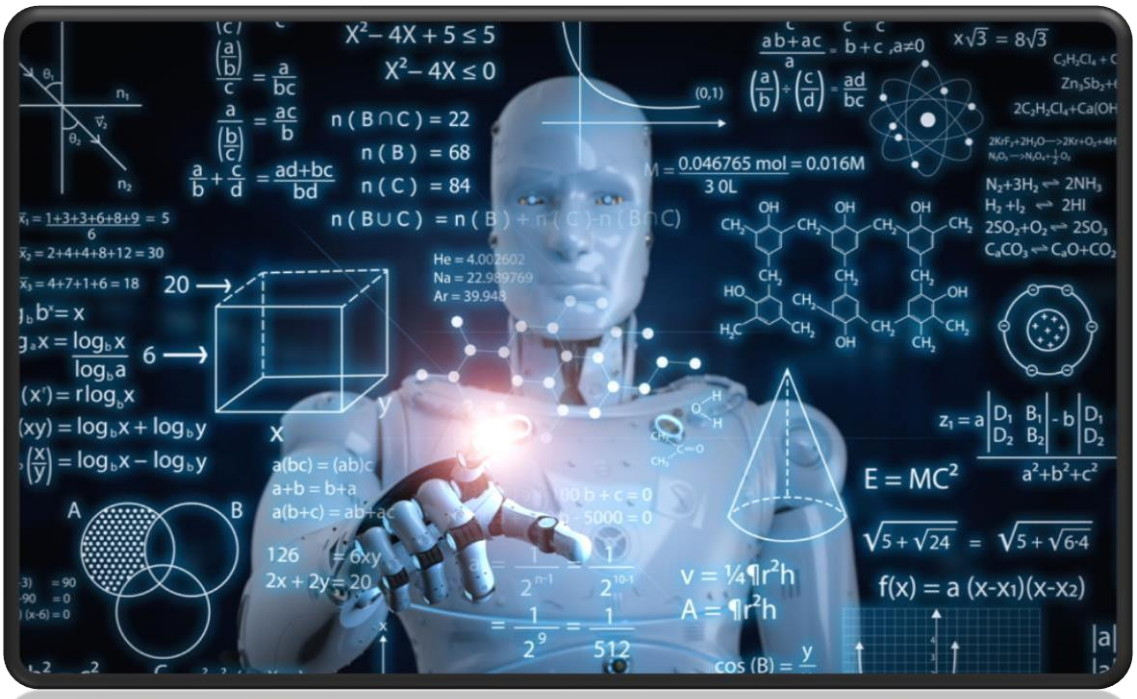


AI in Software Engineering

(CSSE-509)

Name: Wajiha Hanif Arain

Seat no: B18158064

Course Instructor: Dr Humera Tariq

Department of Computer Science

Index Sheet of AI in SE 2021

Course Code: BSCS- 509 Course Tittle: AI in SE Semester: III

S. No	Week	Topic and Lab Practice	Detail	Email	Date
1	Week 1	Machine setup And run code of (Effort estimation)	Run on Jupiter notebook and connected to csv dataset excel file	-	2-3-21
2	Week 1	Linear regression	Solve step by step	-	4-3-21
3	Week 2	Run desharnian.csv dataset on your machine	Run successful after little problems	humera.tariq.dcs.uok.@gmail.com	14-3-21
4	Week 3	Technique for plotting high dimensional data	Write detail about each topic and study individual algorithms	humera.tariq.dcs.uok.@gmail.com	14-3-21
5	Week 4, 5	Effort estimation quiz	Senior's project display	Hardcopy submission	6-4-21
6	Week 6	Train model on dog breed identifier	https://www.kaggle.com/c/dog-breed-identification/code	humera.tariq.dcs.uok.@gmail.com	
7	Week 7	Mid-term	Online submission	humera.tariq.dcs.uok.@gmail.com	27-4-21
8	Week 8	Crunchbase dataset Resend mid-term	Group work	humera.tariq.dcs.uok.@gmail.com	7-5-21
9	Week 9	5 picture-based assignment	Pectoral assignment and have to write AI perspective	-	31-5-21
10	Week 11	Anova table & confusion matrix	Handwritten assignment		15-6-21
11	Week 12	normalization	Handwritten assignment		25-6-21

Week: 01

1. Machine setup for running the lab on Jupiter notebook and prepare Read Me for “Estimating Software Effort using linear regression”.

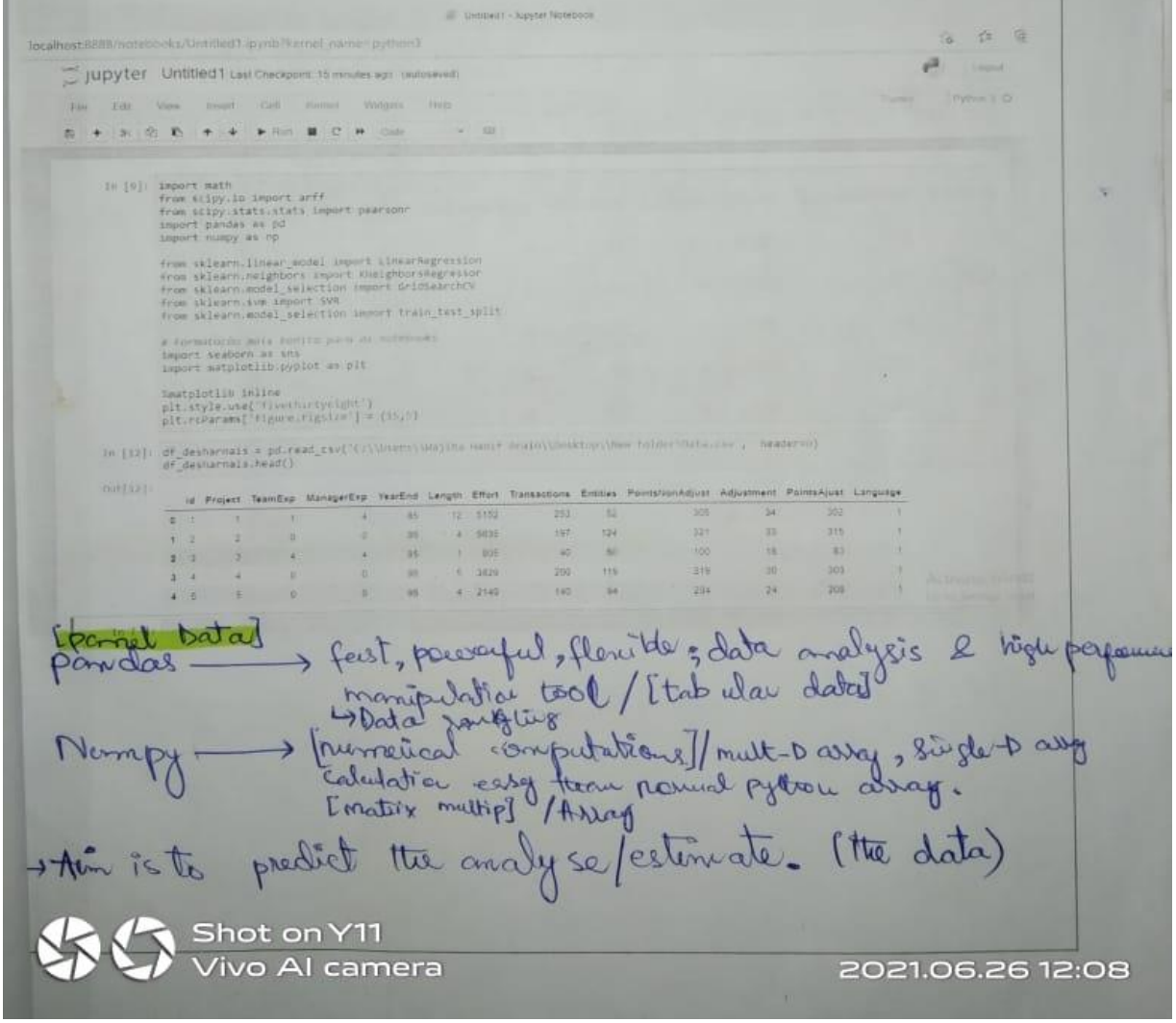
2. “A predictive analysis approach using linear regression to estimate software Efforts” brief explanation and discussion.

3. Derivation of Linear Regression

Read me of: paper I

Estimating software effort using linear regression

- First, download the Anaconda Jupyter in your system.
- Copied that code, from given link & run in jupyter notebook.
- And then, copied the Dataset from that link and paste in your computer's Excel sheet and save it as Data.csv (csv= comma separated value)
- After that in the read_csv, paste that link of Data.csv according to your computer's location than run the code. eg:
  read_csv('C:\\Users\\Wajiha Hanif Arain\\Desktop\\New folder\Data.csv')



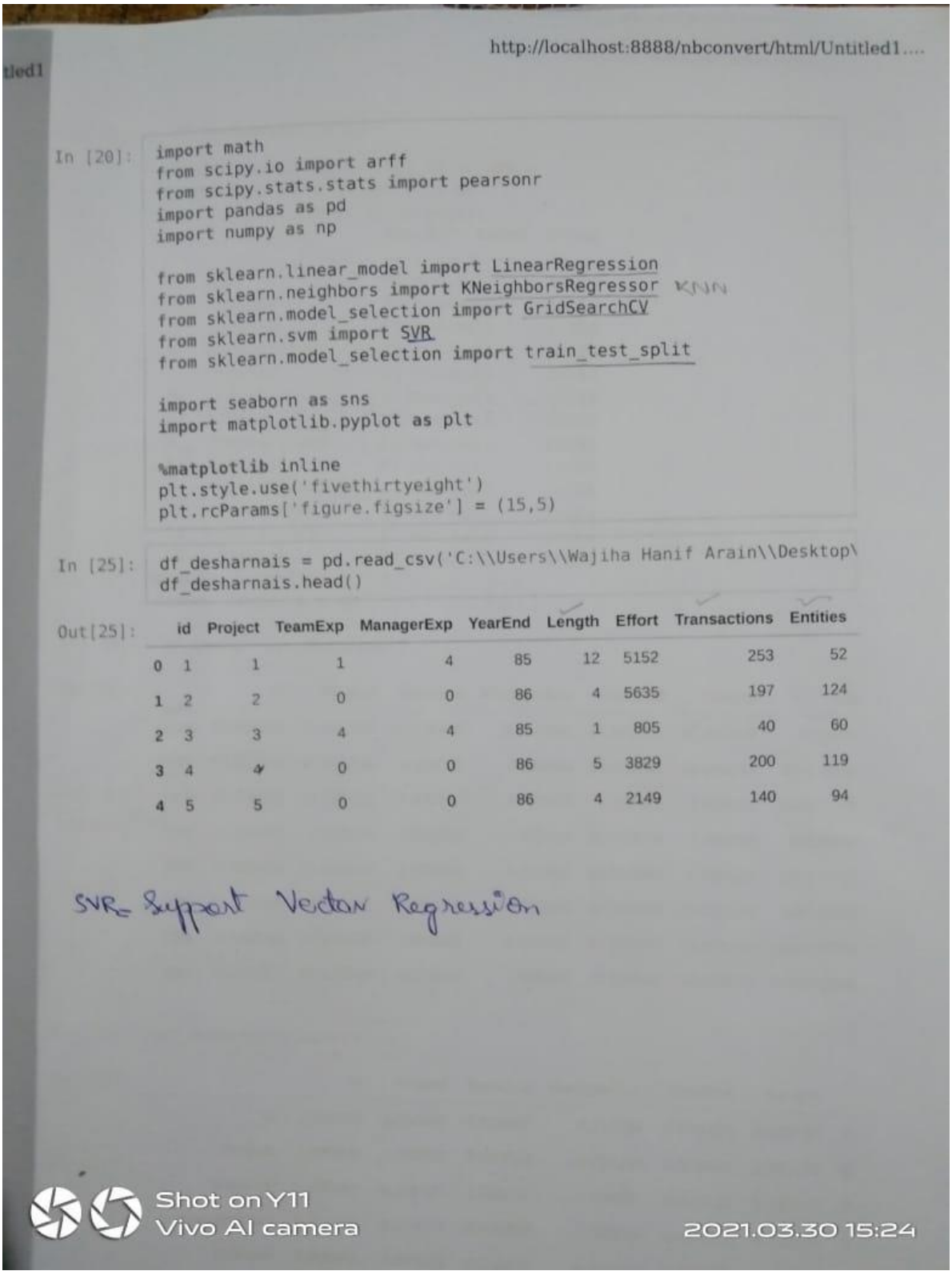
[kernel data]

pandas → fast, powerful, flexible; data analysis & high performance manipulation tool / [tabular data]
↳ Data handling

Numpy → [numerical computations] // multi-D array, single-D array
Calculation easy than normal python array.
[matrix multip] / Array

→ Aim is to predict the analyse/estimate. (the data)

Visualization of Desharnais Dataset



tled1 http://localhost:8888/nbconvert/html/Untitled1....

In [20]:
```
import math
from scipy.io import arff
from scipy.stats.stats import pearsonr
import pandas as pd
import numpy as np

from sklearn.linear_model import LinearRegression
from sklearn.neighbors import KNeighborsRegressor
from sklearn.model_selection import GridSearchCV
from sklearn.svm import SVR
from sklearn.model_selection import train_test_split

import seaborn as sns
import matplotlib.pyplot as plt

%matplotlib inline
plt.style.use('fivethirtyeight')
plt.rcParams['figure.figsize'] = (15,5)
```

In [25]:
```
df_desharnais = pd.read_csv('C:\\Users\\Wajiha Hanif Arain\\Desktop\
df_desharnais.head()
```

Out[25]:

	id	Project	TeamExp	ManagerExp	YearEnd	Length	Effort	Transactions	Entities
0	1	1	1	4	85	12	5152	253	52
1	2	2	0	0	86	4	5635	197	124
2	3	3	4	4	85	1	805	40	60
3	4	4	0	0	86	5	3829	200	119
4	5	5	0	0	86	4	2149	140	94

SVR- Support Vector Regression

In [22]:
```
df_desharnais.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 81 entries, 0 to 80
Data columns (total 13 columns):
 #   Column           Non-Null Count  Dtype
---  ------           --------------  -----
 0   id               81 non-null     int64
 1   Project          81 non-null     int64
 2   TeamExp          81 non-null     int64
 3   ManagerExp       81 non-null     int64
 4   YearEnd          81 non-null     int64
 5   Length           81 non-null     int64
 6   Effort           81 non-null     int64
 7   Transactions     81 non-null     int64
 8   Entities         81 non-null     int64
 9   PointsNonAdjust  81 non-null     int64
 10  Adjustment       81 non-null     int64
 11  PointsAjust      81 non-null     int64
 12  Language         81 non-null     int64
dtypes: int64(13)
memory usage: 8.4 KB
```

In [26]:
```
df_desharnais.describe()
```

Out[26]:

	id	Project	TeamExp	ManagerExp	YearEnd	Length	Effc
count	81.000000	81.000000	81.000000	81.000000	81.000000	81.000000	81.0000
mean	41.000000	41.000000	2.185185	2.530864	85.740741	11.666667	5046.3086
std	23.526581	23.526581	1.415195	1.643825	1.222475	7.424621	4418.7672
min	1.000000	1.000000	-1.000000	-1.000000	82.000000	1.000000	546.0000
25%	21.000000	21.000000	1.000000	1.000000	85.000000	6.000000	2352.0000
50%	41.000000	41.000000	2.000000	3.000000	86.000000	10.000000	3647.0000
75%	61.000000	61.000000	4.000000	4.000000	87.000000	14.000000	5922.0000
max	81.000000	81.000000	4.000000	7.000000	88.000000	39.000000	23940.0000

In [27]:
```
df_desharnais.corr()
```

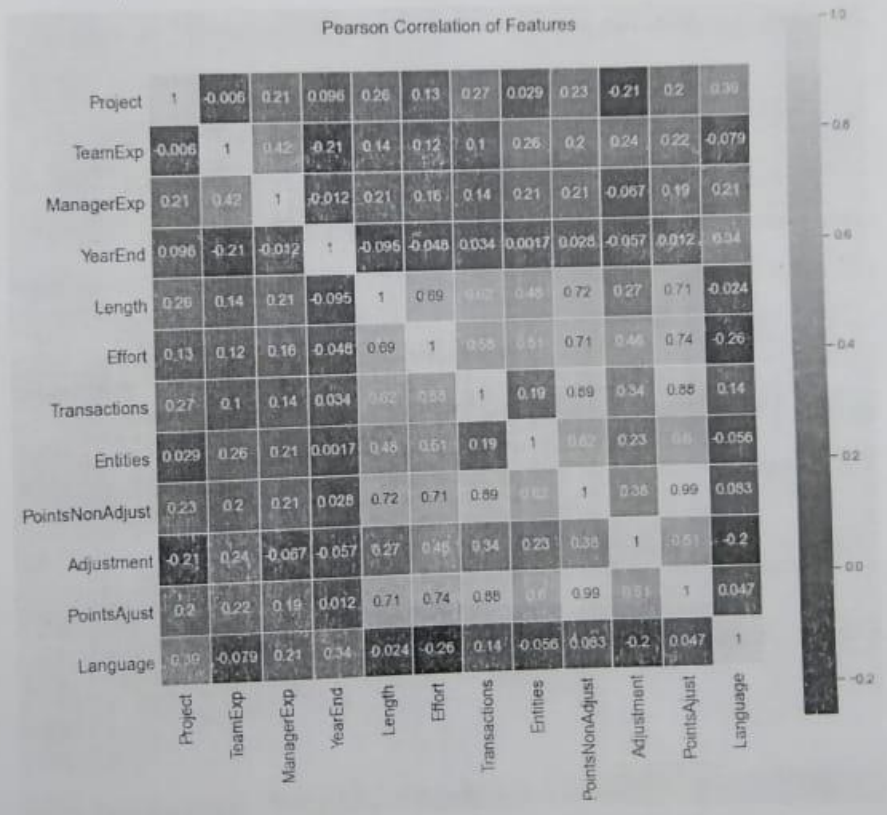
Out[27]:

	id	Project	TeamExp	ManagerExp	YearEnd	Length	
id	1.000000	1.000000	-0.006007	0.214294	0.096486	0.255187	0.
Project	1.000000	1.000000	-0.006007	0.214294	0.096486	0.255187	0.
TeamExp	-0.006007	-0.006007	1.000000	0.424687	-0.210335	0.143948	0.
ManagerExp	0.214294	0.214294	0.424687	1.000000	-0.011519	0.211324	0.
YearEnd	0.096486	0.096486	-0.210335	-0.011519	1.000000	-0.095027	-0.
Length	0.255187	0.255187	0.143948	0.211324	-0.095027	1.000000	0.
Effort	0.126153	0.126153	0.119529	0.158303	-0.048367	0.693280	1.

	id	Project	TeamExp	ManagerExp	YearEnd	Length	
Transactions	0.265891	0.265891	0.103768	0.138146	0.034331	0.620711	0.
Entities	0.028787	0.028787	0.256608	0.206644	0.001686	0.483504	0.
PointsNonAdjust	0.226076	0.226076	0.203805	0.207748	0.028234	0.723849	0.
Adjustment	-0.207774	-0.207774	0.235629	-0.066821	-0.056743	0.266086	0.
PointsAjust	0.202608	0.202608	0.222884	0.187399	0.012106	0.714092	0.

In [28]:
```
colormap = plt.cm.viridis
plt.figure(figsize=(10,10))
plt.title('Pearson Correlation of Features', y=1.05, size=15)
sns.set(font_scale=1.05)
sns.heatmap(df_desharnais.drop(['id'], axis=1).astype(float).corr(),
```

Out[28]: <AxesSubplot:title={'center':'Pearson Correlation of Features'}>



In [30]:
```
features = [ 'TeamExp', 'ManagerExp', 'YearEnd', 'Length', 'Transact
        'PointsNonAdjust', 'Adjustment', 'PointsAjust']

max_corr_features = ['Length', 'Transactions', 'Entities','PointsNor

X = df_desharnais[max_corr_features]
y = df_desharnais['Effort']
```

In [31]:
```
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=

neigh = KNeighborsRegressor(n_neighbors=3, weights='uniform')
neigh.fit(X_train, y_train)
print(neigh.score(X_test, y_test))
```

KNN

```
0.7379861869550943
```

In [32]:
```
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=

model = LinearRegression()
model.fit(X_train, y_train)
print(model.score(X_test, y_test))
```

LR

```
0.7680074954440711
```

In [34]:
```
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=

parameters = {'kernel':('linear', 'rbf'), 'C':[1,2,3,4,5,6,7,8,9,10]

svr = SVR()
LinearSVC = GridSearchCV(svr, parameters, cv=3)
LinearSVC.fit(X_train, y_train)
print("Best params hash: {}".format(LinearSVC.best_params_))
print(LinearSVC.score(X_test, y_test))
```

SVR

```
Best params hash: {'C': 1, 'gamma': 'auto', 'kernel': 'linear'}
0.735919788126071
```

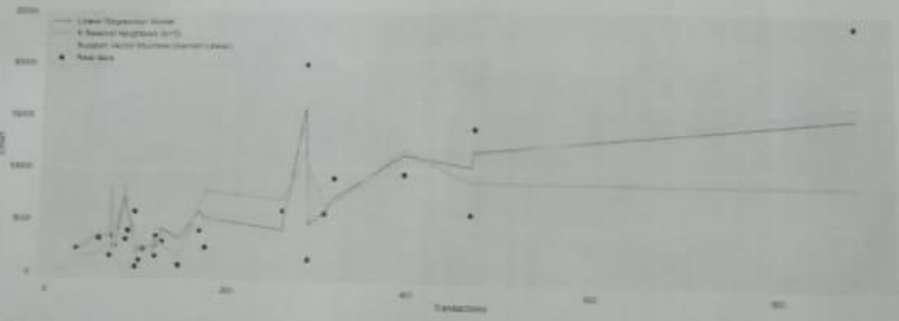
→ LR + KNN applied

→ training of regressors models were performed on 67% of instances.

KNN :
→ predict, the numerical target based on similarity measure.
→ using euclidean formula for d/s measure.
→ KNN motivated by absence of data in Desharnais dataset
→ use 3 neighbours.

LR :-
• aims to varify existence of relationship b/w variables, with one o

In [36]:
```
for i, feature in enumerate(max_corr_features):
    plt.figure(figsize=(18,6))

    # Knn Regression Model
    xs, ys = zip(*sorted(zip(X_test[feature], neigh.fit(X_train, y_t

    # Linear Regression Model
    model_xs, model_ys = zip(*sorted(zip(X_test[feature], model.fit(

    # Support Vector Machine
    svc_model_xs, svc_model_ys = zip(*sorted(zip(X_test[feature], Li

    plt.scatter(X_test[feature], y_test, label='Real data', lw=2,alp
    plt.plot(model_xs, model_ys , lw=2, label='Linear Regression Moc
    plt.plot(xs, ys , lw=2,label='K Nearest Neighbors (k=3)', c='yel
    plt.plot(svc_model_xs, svc_model_ys , lw=2,label='Support Vector

    plt.xlabel(feature)
    plt.ylabel('Effort')
    plt.legend()
    plt.show()
```



A predictive analysis approach using linear regression to estimate software effort

Antogio G. L. Esteves[1], Leonardo M. Medeiros (Orientador)[1]

[1]Pós-graduação em Gerenciamento e Desenvolvimento Ágil de Software
Instituto Federal de Alagoas (IFAL) – Campus Maceió – Maceió - AL – Brasil

toni.esteves@gmail.com, leonardomelomedeiros@gmail.com

Abstract. *Making decisions with a highly uncertain level is a critical problem in the area of software engineering. Predicting software quality requires high accurate tools and high-level experience. AI-based predictive models, on the other hand, are useful tools with an accurate degree that help to make decisions learning from past data. In this study, we build a software effort estimation model to predict the effort before the project development lifecycle, using a linear regression model and also using non-parametric validation model through a Knn regression algorithm.*

1. INTRODUCTION

Software development involves a number of interrelated factors which affect development effort and productivity . The most significant activity in software engineer is the development of projects within the confined timeframe and budget. So accuracy has a vital role for software development, effort prediction estimation is one of the critical tasks required for developing software. In this work our research focus on analyzing the importance of attributes in estimating software cost as well as its correlation.

In this paper, we set out to answer two research questions related to the dataset:

1. Which the correlation of each metrics in the estimation of software effort ?
2. How accurate is the model of software effort ?

2. EFFORT ESTIMATION

When measurements embrace structure system they become more meaningful indicators called metrics. Metrics are conceived by the user and designed to reveal chosen characteristics in a reliable meaningful manner. Then these metrics are mapped to ongoing measurements, to arrive at a best fit [Pandian 2003]

One of the fundamental issues in a software project is to know, before executing it, how much effort, in working hours, it will be necessary to bring it to term. This area called effort estimation counts on some techniques that have presented interesting results over the last few years[Wazlawick 2013].

One of reasons for failed estimations is an insufficient background of information in the area of software estimation. Unfortunately, human experts are not always as good at estimating as one could hope: estimates of cost and effort in software projects are often [illegible] average overrun of about 30% [Halkjelsvik and Jørgensen 2011].

Deliberate decisions regarding the particular estimation method and knowledgeable use require insight into the principles of effort estimation[Trendowicz and Jeffery 2014].

[*Learning-oriented models* attempt to automate the estimation process by building computerised models that can learn from previous estimation experience [Boehm et al. 2000]. These models do not rely on assumptions and are capable of learning incrementally as new data are provided over time][Lee-Post et al. 1998].

2.1. RELATED WORKS

The research developed by Ayyıldız makes use of Desharnais dataset to finding the necessary attributes that affects the software effort estimation and analyzing the necessity of these attributes [Erçelebi Ayyıldız and Can Terzi 2017][The Pearson's Correlation correlations between metrics of Desharnais dataset and software effort are analyzed and applicability of the regression analysis is examined.]

To show the differences between the actual and estimated values of the dependent variable, prediction performance are evaluated using Magnitude of Relative Error (MRE), Mean Magnitude of Relative Error (MMRE), Median Magnitude of Relative Error (MdMRE), MSE (Mean Square Error) and Prediction Quality (pred(e)).

One of the most complete studies was presented by Kitchenham [Kitchenham et al. 2002]. In her study, was present a data set that enables to investigate the actual accuracy of industrial estimates and to compare those estimates with estimates produced from various function point estimation models. However, the study make it clear that any models derived from the current data set are context-specific. The conclusions drawed from this study are somewhat limited, because the projects studied were undertaken by a single company. Thus, it was not expected any of the models presented in this paper to generalize automatically to other maintenance or development situations.

3. MATERIALS AND METHODS

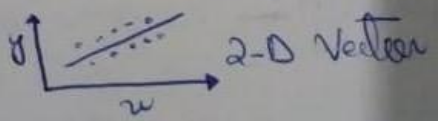
To perform our study firstly we analyze the correlation between each attributes of Desharnais dataset and effort attribute. We apply linear regression technique to investigate relation between these attributes. After that we apply a regression based on k-nearest neighbors regressor. Lastly we evaluate our prediction performance comparing the squared error value of both algorithms .

3.1. DATASET

To perform this study we used Desharnais dataset [1] which is composed of a total of 81 projects developed by a Canadian software house in 1989. This data set includes nine numerical attributes. The eight independent attribute of this data set, namely "TeamExp", "ManagerExp", "YearEnd", "Length", "Transactions", "Entities", "PointsAdj", and "PointsNonAjust" are all considered for constructing the models. The dependent attribute "Effort" is measured in person hours.

[1]The promise repository of empirical software engineer data.

All x → independent



3.2. FEATURE SELECTION

To address Desharnais dataset the correlations between attributes and software effort are analyzed. The correlation between two variables is a measure of how well the variables are related.A feature is an individual measurable property of the process being observed.

The most common measure of correlation in statistics is the Pearson Correlation Pearson correlation coefficient (PCC), wich is a statistical metric that measures the strength and direction of a linear relationship between two random variables [Rodgers and Nicewander 1988]. Pearson correlation coefficient analysis produces a result between -1 and 1. Results between 0.5 and 1.0 indicate high correlation [Mehedi Hassan Onik et al. 2018]. The Pearson correlation coefficients between attributes and software efforts are given in Figure 1 for Desharnais dataset.

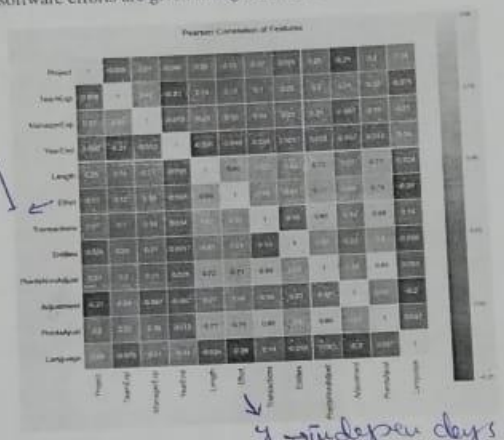


Figure 1. Pearsons Correlation to Desharnais dataset

3.3. MODELS CONSTRUCTION

In this study the following algorithms were used: *Linear Regression* and *K-Nearest Neighbors Regression*. The training of the models was carried out in Python language, along with the following libraries: Numpy, Pandas, Scikit-learn, Seaborn and Matplotlib. During the training it was necessary to estimate the values of the random state parameter, since they are not previously known.

The regression analysis aims to verify the existence of a functional relationship between a variable with one or more variables, obtaining an equation that explains the variation of the dependent variable Y, by the variation of the levels of the independent variables. The training of the *Linear Regression* model consists of generating a regression for the target variable Y. Thus a linear regression line has an equation of the form $Y = a + bX$, where X is the explanatory variable and Y is the dependent variable. The slope of the line is b, and a is the intercept (the value of y when $x = 0$).

Likewise the *K-Nearest Neighbor Regression* is a simple algorithm that stores all available cases and predict the numerical target based on a similarity measure and it's been used in a statistical estimation and pattern recognition as non-parametric technique classifying correctly unknown cases calculating euclidean distance between data points. In fact our choice by *K-Nearest Neighbor Regression* was motivated by the absence of a detailed explanation about how effort attribute value is calculated on Desharnais dataset.

4. RESULTS

Both models generated from the training with data from the previous section will be applied to the remaining 33% of the base, previously isolated, and their performances will be evaluated in order to demonstrate how accurate the linear regression model can predict software effort estimation. Thus, we calculate respective R^2 values. Table 3 shows the coefficients reached.

Algorithm	R^2 Score
Linear Model Regression	0.7680074954440712
K-Nearest Neighbor Regressor	0.7379861869550943

Table 1: Algorithms model results

In Figure 2 plots of the best correlated variables applied to both models are displayed.

(a) *Knn* x *LR* on Length feature

(b) *Knn* x *LR* on Entities feature

(c) *Knn* x *LR* on PointsAdjust feature

(d) *Knn* x *LR* on PointsNonAdjust feature

(e) *Knn* x *LR* on Transactions feature

Figure 2. Comparative R^2 scores from K-neighbors Regression and Linear Regression

Each feature from more correlated features is illustrated in Figure 2. The figure shows the linear model (blue line) prediction is fairly close to Knn model effort prediction (red line), predicting the numerical target based on a similarity measure.

5. CONCLUSION AND FUTURE WORKS

The contributions of this work are based on the use of two output models that seek to take advantage of the relationships between the target values of the project. These methods, together with linear regression and K-neighbors regression algorithms , resulted in predictive models capable of estimating values for the software effort estimation operations. The results of our empirical study reveal that predictive model of software effort presented by both models, could successfully predict more than 70% with less than 3% difference between them.

Our results obtained obtained a R^2 value of more than 70% and a difference of only 3% among them, indicating the feasibility of using linear regressors to predict software effort. However, to have a more concise and fair result we need to reproduce the same approach with other available algorithms.

Finally, we propose as future works the use of a larger project base in order to diversify and give greater reliability to the method. Another point to consider is to apply these models in order to compare them with the function points.

References

Boehm, B., Abts, C., and Chulani, S. (2000). Software development cost estimation approaches – a survey. *Ann. Softw. Eng.*, 10(1-4):177–205.

Erçelebi Ayyıldız, T. and Can Terzi, H. (2017). Case study on software effort estimation. 7:103–107.

Halkjelsvik, T. and Jørgensen, M. (2011). From origami to software development: A review of studies on judgment-based predictions of performance time. 138:238–71.

Kitchenham, B., Pfleeger, S. L., McColl, B., and Eagan, S. (2002). An empirical study of maintenance and development estimation accuracy. *Journal of Systems and Software*, 64(1):57 – 77.

Lee-Post, A., Cheng, C. H., and Balakrishnan, J. (1998). Software development cost estimation: Integrating neural network with cluster analysis. 34:1–9.

Mehedi Hassan Onik, M., Ahmmed Nobin, S., Ferdous Ashrafi, A., and Mohmud Chowdhury, T. (2018). Prediction of a Gene Regulatory Network from Gene Expression Profiles With Linear Regression and Pearson Correlation Coefficient. *ArXiv e-prints*.

Pandian, C. R. (2003). Software metrics: A guide to planning, analysis, and application.

Rodgers, J. and Nicewander, W. (1988). Thirteen ways to look at the correlation coefficient. *The American Statistician*, 42(1):59–66.

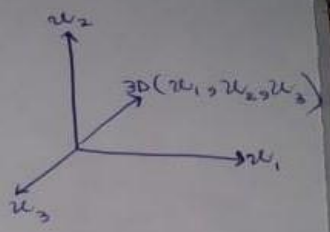
Trendowicz, A. and Jeffery, R. (2014). *Software Project Effort Estimation: Foundations and Best Practice Guidelines for Success*. Springer Publishing Company, Incorporated.

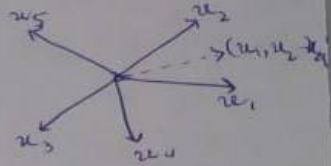
Wazlawick, R. (2013). *ENGENHARIA DE SOFTWARE: CONCEITOS E PRÁTICAS*. Elsevier Editora Ltda.

To address Deshamais dataset the correlations between attributes and software effort are analyzed. The correlation between two variables is a measure of

Variables/attributes Reducing / feature engineering

↳ Linear regression is a straight line





High Dimension

↳ In computer science, usually points treated as vector.

$[x_1 \quad x_2 \quad x_3]$

- ↳ Linear Regression
- ↳ Knn
- ↳ Metrics
- ↳ Best fit
- ↳ effort estimation
- ↳ cost/effort

- ↳ Deshamais dataset
- ↳ Pearson's corr.

Linear Regression:

In statistics, linear regression is linear approach to modelling relationship b/w scalar response & one or more explanatory var [also known as dependent & independent var]

→ scatterplot → fitting line [no correlation b/w x & y]

→ $Y = a + bX$

- X ⇒ explanatory var.
- Y ⇒ dependent var.
- b ⇒ slope
- a ⇒ intercept. ∴ $y = x = 0$

↳ correlation

↳ outlier → few points which lie far away from main cluster

Sometimes it reflects anomalous data

- If point lies far from other data in horizontal direction, it known as influential observation
- Residuals / Error

KNN Regression:

↳ it's non-parametric method, in intuitive manner approx. association b/w ind. var(s) & contn. outcome by averaging observation in same neighbourhood.

↳ Size of neighbourhood selected by analyst

↳ use in classification problems.

→ methods to calculate d/s b/w points.
eg 11 people with certain height, age, weight
here weight (target)
• ID, gives all data, but ID11 weight empty
but age & height given, so predict by
some neighbour nature.
• choose closest data
→ Methods to calculate d/s b/w points.
→ Euclidean distance → $\sqrt{\sum_{i=1}^{k} [x_i - y_i]^2}$
→ Manhattan Distance (cont.)
→ Hamming distance (categorical → $\sum_{i=1}^{k} |x_i - y_i|$

Effort Estimation:

process of predicting most realistic amount of effort (expressed in terms of person-hrs or money)

Pearsons Correlation:

This is best statistics that measure statistical relationship, or association b/w two cont. var.
→ Gives info. about magnitude of association or correlation as well as direction of relationship

{relation b/w → response var & explanatory var}

Data Analysis Toolkit #10: Simple linear regression Page 1

Simple linear regression is the most commonly used technique for determining how one variable of interest (the response variable) is affected by changes in another variable (the explanatory variable). The terms "response" and "explanatory" mean the same thing as "dependent" and "independent", but the former terminology is preferred because the "independent" variable may actually be interdependent with many other variables as well.

Simple linear regression is used for three main purposes:

1. To describe the linear dependence of one variable on another
2. To predict values of one variable from values of another, for which more data are available
3. To correct for the linear dependence of one variable on another, in order to clarify other features of its variability.

Any line fitted through a cloud of data will deviate from each data point to greater or lesser degree. The vertical distance between a data point and the fitted line is termed a "residual". This distance is a measure of prediction error, in the sense that it is the discrepancy between the actual value of the response variable and the value predicted by the line. Linear regression determines the best-fit line through a scatterplot of data, such that the sum of squared residuals is minimized; equivalently, it minimizes the error variance. The fit is "best" in precisely that sense: the sum of squared errors is as small as possible. That is why it is also termed "Ordinary Least Squares" regression.

ŷ = predicted value y = actual value

Derivation of linear regression equations

The mathematical problem is straightforward:

- given a set of n points (X_i, Y_i) on a scatterplot,
- find the best-fit line, $\hat{Y} = a + bX$,
- such that the sum of squared errors in Y, $\sum (Y_i - \hat{Y}_i)^2$, is minimized

The derivation proceeds as follows. For convenience, name the sum of squares "Q":

$$Q = \sum_{i=1}^{n} [Y_i - \hat{Y}] = \sum_{i=1}^{n} (Y_i - a - bX_i)^2 \quad (1)$$

Then, Q will be minimized at the values of a and b for which $\partial Q / \partial a = 0$ and $\partial Q / \partial b = 0$. The first of these conditions is:

$$\frac{\partial Q}{\partial a} = \sum_{i=1}^{n} -2(Y_i - a - bX_i) = 2\left(na + b\sum_{i=1}^{n} X_i - \sum_{i=1}^{n} Y_i\right) = 0 \quad (2)$$

which, if we divide through by 2 and solve for a, becomes simply:

$$a = \bar{Y} - b\bar{X} \quad (3)$$

which says that the constant a (the y-intercept) is set such that the line must go through the mean of x and y. This makes sense, because this point is the "center" of the data cloud. The second condition for minimizing Q is:

$$\frac{\partial Q}{\partial b} = \sum_{i=1}^{n} -2X_i(Y_i - a - bX_i) = \sum_{i=1}^{n} -2(X_i Y_i - aX_i - bX_i^2) = 0 \quad (4)$$

If we substitute the expression for a from (3) into (4), then we get:

$$\sum_{i=1}^{n} (X_i Y_i - X_i\bar{Y} + b\bar{X}X_i - bX_i^2) = 0 \quad (5)$$

We can separate this into two sums:

$$\sum_{i=1}^{n} (X_i Y_i - X_i\bar{Y}) - b\sum_{i=1}^{n} (X_i^2 - X_i\bar{X}) = 0$$

which becomes directly:

$$b = \frac{\sum_{i=1}^{n} [X_i Y_i - X_i\bar{Y}]}{\sum_{i=1}^{n} [X_i^2 - X_i\bar{X}]} \quad (6)$$

Data Analysis Toolkit #10: Simple linear regression

$$b = \frac{\sum_{i=1}^{n}(X_i Y_i) - n\bar{X}\bar{Y}}{\sum_{i=1}^{n}(X_i^2) - n\bar{X}^2} \qquad (7)$$

We can translate (7) into a more intuitively obvious form, by noting that

$$\sum_{i=1}^{n}(\bar{X}^2 - X_i\bar{X}) = 0 \quad \text{and} \quad \sum_{i=1}^{n}(\bar{X}\bar{Y} - Y_i\bar{X}) = 0 \qquad (8)$$

so that b can be rewritten as the ratio of Cov(x,y) to Var(x)

$$b = \frac{\sum_{i=1}^{n}(X_i Y_i - X_i\bar{Y}) + \sum_{i=1}^{n}(\bar{X}\bar{Y} - Y_i\bar{X})}{\sum_{i=1}^{n}(X_i^2 - X_i\bar{X}) + \sum_{i=1}^{n}(\bar{X}^2 - X_i\bar{X})} = \frac{\frac{1}{n}\sum_{i=1}^{n}(X_i - \bar{X})(Y_i - \bar{Y})}{\frac{1}{n}\sum_{i=1}^{n}(X_i - \bar{X})^2} = \frac{Cov(X,Y)}{Var(X)} \qquad (9)$$

The quantities that result from regression analyses can be written in many different forms that are mathematically equivalent but superficially distinct. All of the following forms of the regression slope b are mathematically equivalent:

$$b = \frac{Cov(X,Y)}{Var(X)} \text{ or } \frac{\sum xy}{\sum x^2} \text{ or } \frac{\sum_{i=1}^{n}(X_i Y_i) - \frac{\sum_{i=1}^{n}X_i \sum_{i=1}^{n}Y_i}{n}}{\sum_{i=1}^{n}(X_i)^2 - \frac{\left(\sum_{i=1}^{n}X_i\right)^2}{n}} \text{ or } \frac{\sum_{i=1}^{n}(X_i Y_i) - n\bar{X}\bar{Y}}{\sum_{i=1}^{n}(X_i^2) - n\bar{X}^2} \text{ or } \frac{\frac{1}{n}\sum_{i=1}^{n}(X_i Y_i) - \bar{X}\bar{Y}}{\frac{1}{n}\sum_{i=1}^{n}(X_i^2) - \bar{X}^2} \text{ or } \frac{\overline{(XY)} - \bar{X}\bar{Y}}{\overline{(X_i^2)} - \bar{X}^2} \qquad (10)$$

A common notational shorthand is to write the "sum of squares of X" (that is, the sum of squared deviations of the X's from their mean), the "sum of squares of Y", and the "sum of XY cross products" as,

$$\sum x^2 = SS_x = (n-1)Var(X) = \sum_{i=1}^{n}(X_i - \bar{X})^2 = \sum_{i=1}^{n}(X_i^2) - n\bar{X}^2 \qquad (11)$$

$$\sum y^2 = SS_y = (n-1)Var(Y) = \sum_{i=1}^{n}(Y_i - \bar{Y})^2 = \sum_{i=1}^{n}(Y_i^2) - n\bar{Y}^2 \qquad (12)$$

$$\sum xy = S_{xy} = (n-1)Cov(X,Y) = \sum_{i=1}^{n}(X_i - \bar{X})(Y_i - \bar{Y}) = \sum_{i=1}^{n}(X_i Y_i) - n\bar{X}\bar{Y} \qquad (13)$$

It is important to recognize that Σx^2, Σy^2, and Σxy, as used in Zar and in equations (10)-(13), are not summations; instead, they are *symbols* for the sums of squares and cross products. Note also that S and SS in (11)-(13) are uppercase S's rather than standard deviations.

Besides the regression slope b and intercept a, the third parameter of fundamental importance is the correlation coefficient r or the coefficient of determination r^2. r^2 is the ratio between the variance in Y that is "explained" by the regression (or, equivalently, the variance in $\hat{Y}$), and the total variance in Y. Like b, r^2 can be calculated many different ways.

$$r^2 = \frac{Var(\hat{Y})}{Var(Y)} = \frac{b^2 Var(X)}{Var(Y)} = \frac{[Cov(x,y)]^2}{Var(X)Var(Y)} = \frac{Var(Y) - Var(Y - \hat{Y})}{Var(Y)} = \frac{S_{xy}^2}{SS_x SS_y} \qquad (14)$$

* If points are at large distance then their variance is greater.

* Problem statement:- Exactly telling what you're minimizing, what are you maximizing & how have you designed the problem

• variance → Similarity matching → correlating

2) To maximize the variance b/c we're trying to Similarity matching b/w vectors

Equation (14) implies the following relationship between the correlation coefficient, r, the regression slope, b, and the standard deviations of X and Y (s_X and s_Y):

$$r = b\frac{S_x}{S_y} \quad \text{and} \quad b = r\frac{S_y}{S_x} \tag{15}$$

The residuals e_i are the deviations of each response value Y_i from its estimate $\hat{Y}_i$. These residuals can be summed in the sum of squared errors (SSE). The mean square error (MSE) is just what the name implies, and can also be considered the "error variance" ($s^2_{Y\cdot X}$). The root-mean-square-error (RMSE), also termed the "standard error of the regression" ($s_{Y\cdot X}$) is the standard deviation of the residuals. The mean square error and RMSE are calculated by dividing by n-2, because linear regression removes two degrees of freedom from the data (by estimating two parameters, a and b).

$$e_i = Y_i - \hat{Y}_i SSE = \sum_{i=1}^{n} e_i^2 MSE = s^2_{Y\cdot X} = \frac{SSE}{n-2} = Var(Y)(1-r^2)\frac{n-1}{n-2} RMSE = s_{Y\cdot X} = \sqrt{\frac{SSE}{n-2}} = s_Y\sqrt{\frac{n-1}{n-2}}\sqrt{1-r^2} \tag{16}$$

where Var(Y) is the sample, not population, variance of Y, and the factors of n-1/n-2 serve only to correct for changes in the number of degrees of freedom between the calculation of variance (d.f.=n-1) and $s_{Y\cdot X}$ (d.f.=n-2).

Uncertainty in regression parameters

The standard error of the regression slope b can be expressed many different ways, including:

$$s_b = \sqrt{\frac{SS_Y/SS_X - b^2}{n-2}} = \frac{s_{Y\cdot X}}{\sqrt{SS_X}} = \frac{1}{\sqrt{n}}\frac{s_{Y\cdot X}}{s_X} = \frac{s_Y}{s_X}\frac{\sqrt{1-r^2}}{\sqrt{n-2}} = \frac{b}{r}\frac{\sqrt{1-r^2}}{\sqrt{n-2}} = \frac{b}{\sqrt{n-2}}\sqrt{\frac{1}{r^2}-1} \tag{17}$$

If all of the assumptions underlying linear regression are true (see below), the regression slope b will be approximately t-distributed. Therefore, confidence intervals for b can be calculated as,

$$CI = b \pm t_{\alpha(2),n-2}s_b \tag{18}$$

To determine whether the slope of the regression line is statistically significant, one can straightforwardly calculate t, the number of standard errors that b differs from a slope of zero:

$$t = \frac{b}{s_b} = r\frac{\sqrt{n-2}}{\sqrt{1-r^2}} \tag{19}$$

and then use the t-table to evaluate the α for this value of t (and n-2 degrees of freedom). The uncertainty in the elevation of the regression line at the mean X (that is, the uncertainty in $\bar{Y}$ at the mean X) is simply the standard error of the regression $s_{Y\cdot X}$, divided by the square root of n. Thus the standard error in the predicted value of $\hat{Y}_i$ for some X_i is the uncertainty in the elevation at the mean X, plus the uncertainty in b times the distance from the mean X to X_i, added in quadrature:

$$s_{\hat{Y}_i} = \sqrt{\left(s_{Y\cdot X}/\sqrt{n}\right)^2 + \left(s_b(X_i - \bar{X})\right)^2} = s_{Y\cdot X}\sqrt{\frac{1}{n} + \frac{(X_i - \bar{X})^2}{SS_X}} = \frac{s_{Y\cdot X}}{\sqrt{n}}\sqrt{1 + \frac{(X_i - \bar{X})^2}{Var(X)}} \tag{20}$$

where Var(X) is the population, (not sample) variance of X (that is, it is calculated with n rather than n-1). $\hat{Y}_i$ is also t-distributed, so a confidence interval for $\hat{Y}_i$ can be estimated by multiplying the standard error of $\hat{Y}_i$ by $t_{\alpha(2),n-2}$. Note that this confidence interval grows as X_i moves farther and farther from the mean of X. Extrapolation beyond the range of the data assumes that the underlying relationship continues to be linear beyond that range. Equation (20) gives the standard error of the $\hat{Y}_i$, that is, the Y-values predicted by the regression line. The uncertainty in a new individual value of Y (that is, the prediction interval rather than the confidence interval) depends not only on the uncertainty in where the regression line is, but also the uncertainty in where the individual data point Y lies in relation to the regression line. This latter uncertainty is simply the standard deviation of the residuals, or $s_{Y\cdot X}$, which is added (in quadrature) to the uncertainty in $\hat{Y}_i$, as follows:

$$\left(s_{\hat{Y}_i}\right) = \sqrt{s_{Y\bullet X}^2 + s_{\hat{Y}_i}^2} = s_{Y\bullet X}\sqrt{1 + \frac{1}{n} + \frac{\left(X_i - \overline{X}\right)^2}{SS_X}} \tag{21}$$

The standard error of the Y-intercept, a, is just a special case of (20) for X_i=0,

$$s_a = \sqrt{\left(s_{Y\bullet X}/\sqrt{n}\right)^2 + \left(s_b\overline{X}\right)^2} = s_{Y\bullet X}\sqrt{\frac{1}{n} + \frac{\overline{X}^2}{SS_X}} \tag{22}$$

The standard error of the correlation coefficient r is,

$$s_r = \sqrt{\frac{1-r^2}{n-2}} \tag{23}$$

We can test whether the correlation between X and Y is statistically significant by comparing t to its standard error,

$$t = \frac{r}{s_r} = r\frac{\sqrt{n-2}}{\sqrt{1-r^2}} \tag{24}$$

and looking up this value in a t-table. Note that $t=r/s_r$ has the same value as $t=b/s_b$; that is, the statistical significance of the correlation coefficient r is equivalent to the statistical significance of the regression slope b.

Assumptions behind linear regression

The assumptions that must be met for linear regression to be valid depend on the purposes for which it will be used. Any application of linear regression makes two assumptions:

(A) The data used in fitting the model are representative of the population.

(B) The true underlying relationship between X and Y is linear.

All you need to assume to predict Y from X are (A) and (B). To estimate the standard error of the prediction $s_{\hat{Y}_i}$, you also must assume that:

(C) The variance of the residuals is constant (homoscedastic, not heteroscedastic).

For linear regression to provide the best linear unbiased estimator of the true Y, (A) through (C) must be true, and you must also assume that:

(D) The residuals must be independent.

To make probabilistic statements, such as hypothesis tests involving b or r, or to construct confidence intervals, (A) through (D) must be true, and you must also assume that:

(E) The residuals are normally distributed.

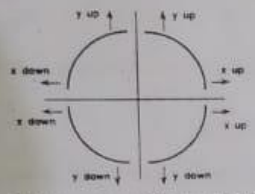
Contrary to common mythology, linear regression does *not* assume *anything* about the distributions of either X or Y; it only makes assumptions about the distribution of the residuals e_i. As with many other statistical techniques, it is not necessary for the data themselves to be normally distributed, only for the errors (residuals) to be normally distributed. And this is only required for the statistical significance tests (and other probabilistic statements) to be valid; regression can be applied for many other purposes even if the errors are non-normally distributed.

Steps in constructing good regression models

1. Plot and examine the data.
2. If necessary, transform the X and/or Y variables so that:
   - the relationship between X and Y is linear, and
   - Y is homoskedastic (that is, the scatter in Y is constant from one end of the X data to the other)

If (as is often the case), the scatter in Y increases with increasing Y, the heteroscedasticity can be eliminated by transforming Y downward on the "ladder of powers" (see the toolkit on transforming distributions). Conversely, if the scatter in Y is greater for smaller Y, transform Y upward on the ladder of powers.

Curvature in the data can be reduced by transforming Y and/or X up or down the ladder of powers according to the "bulging rule" of Mosteller and Tukey (1977), which is illustrated in the following diagram:



The bulging rule for transforming curvature to linearity
(after Mosteller and Tukey, 1977).

Note that transforming X will change the curvature of the data without affecting the variance of Y, whereas transforming Y will affect both the shape of the data and the heteroscedasticity of the data. Note that visual assessments of the "scatter" in the data are vulnerable to an optical illusion: if the data density changes with X, the spread in the Y values will look larger wherever there are more data, even if the error variance is constant throughout the range of X.

3. Calculate the linear regression statistics. Every standard statistics package does this, as do many spreadsheets, pocket calculators, etc. It is not difficult to do by hand (or via a custom spreadsheet), as the example on page 6 illustrates. The steps are as follows:

(a) for each data point, calculate X_i^2, Y_i^2, and X_iY_i
(b) calculate the sums of the X_i, Y_i, X_i^2, Y_i^2, and X_iY_i
(c) calculate the sums of squares SS_X, SS_Y, and S_{XY} (also written Σx^2, Σy^2, and Σxy) via(11)-(13)
(d) calculate a, b, and r^2 via (10), (3), and (14)
(e) calculate s_b and s_r via (17) and (23)

4. (a) Examine the regression slope and intercept. Are they physically plausible? Within a plausible range of X values, does the regression equation predict reasonable values of Y? (b) Does r^2 indicate that the regression explains enough variance to make it useful? "Useful" depends on your purpose: if you seek to predict Y accurately, then you want to be able to explain a substantial fraction of the variance in Y. If, on the other hand, you want to simply clarify how X affects Y, a high r^2 is not important (indeed, part of your task of clarification consists in determining how much of the variation in Y is explainable by variation in X). (c) Does the standard error of the slope indicate that b is precise enough for your purposes? If you want to predict Y from X, are the confidence intervals for Y adequate for your purposes?

Important note: r^2 is often largely irrelevant to the task at hand, and slavishly seeking to obtain the highest possible r^2 is often counterproductive. In polynomial regression or multiple regression, adding more adjustable coefficients to the regression equation will always increase r^2, even though doing so may not improve the predictive validity of the fitted equation. Indeed, it may undermine the usefulness of the analysis, if one begins fitting to the *noise* in the data rather than the *signal*.

5. Examine the residuals, $e_i = Y_i - \hat{Y}_i$. The following residual plots are particularly useful:

5(a). Plot the residuals versus X. (see examples on pp. 7-8)
-If the residuals increase or decrease with X, they are heteroscedastic. Transform Y to cure this.
-If the residuals are curved with X, the relationship between X and Y is nonlinear. Either transform X, or fit a nonlinear curve to the data.
-If there are outliers, check their validity, and/or use robust regression techniques.

5(b) Plot the residuals versus $\hat{Y}$, again to check for heteroscedasticity (this step is redundant with 5(a) for simple one-variable linear regression, and can be skipped).

5(c) Plot the residuals against every other possible explanatory variable in the data set.
-If the residuals are correlated with another variable (call it Z), then check to see whether Z is also correlated with X. If both the residuals and X are correlated with Z, then the regression slope will *not*

accurately reflect the dependence of Y on X. You must either: (1) correct both X and Y for changes in Z, before regressing Y on X, or preferably (2) use multiple regression, or another fitting technique that can account for the interactions between X and Z and their combined effect on Y. If the residuals are correlated with Z, but X is not, then multiple regression on both X and Z will allow you to predict Y more accurately (that is, explain more of the variance in Y), but ignoring Z will not bias the regression slope of Y on X.

5(d) Plot the residuals against time.
-Check for seasonal variation, or long-term trend. Again, they should be accounted for, if they are present.

5(e) Plot the residuals against their lags (that is, plot e_i versus e_{i-1}).
-If the residuals are strongly correlated with their lags, the residuals are serially correlated. Serial correlation (also called autocorrelation) means the true uncertainties in the relationship between X and Y will be larger (potentially *much* larger) than suggested by the calculated standard errors. Dealing with serial correlation requires special techniques such as the Hildreth-Lu procedure, which will be explained in a later toolkit.

5(f) Plot the distribution of the residuals (either as a histogram, or a normal quantile plot)
-If the residuals are not normally distributed, your estimates of statistical significance and confidence intervals will not be accurate.

6 Check for outliers, both visually and statistically (see Helsel and Hirsch, section 9.5 for more information). One of the simplest measures of influence is Cook's "D", calculated for each data point as

$$D_i = \frac{e_i^2}{2s_{Y \bullet X}^2}\left(\frac{1}{n} + \frac{\left(X_i - \overline{X}\right)^2}{SS_X}\right)$$

Points with D_i values greater than $F_{0.1,3,n-2}$ are considered to have a large influence on the regression line; for n greater than about 30 this corresponds to $D_i \approx 2.4$. High influence does not automatically make a point an outlier, but it does mean that it makes a substantial difference whether the point is included or not. If the point can be shown to be a mistake, then it should be either corrected or deleted. If it is not a mistake, or if you are unsure, then a more robust regression technique is often advisable.

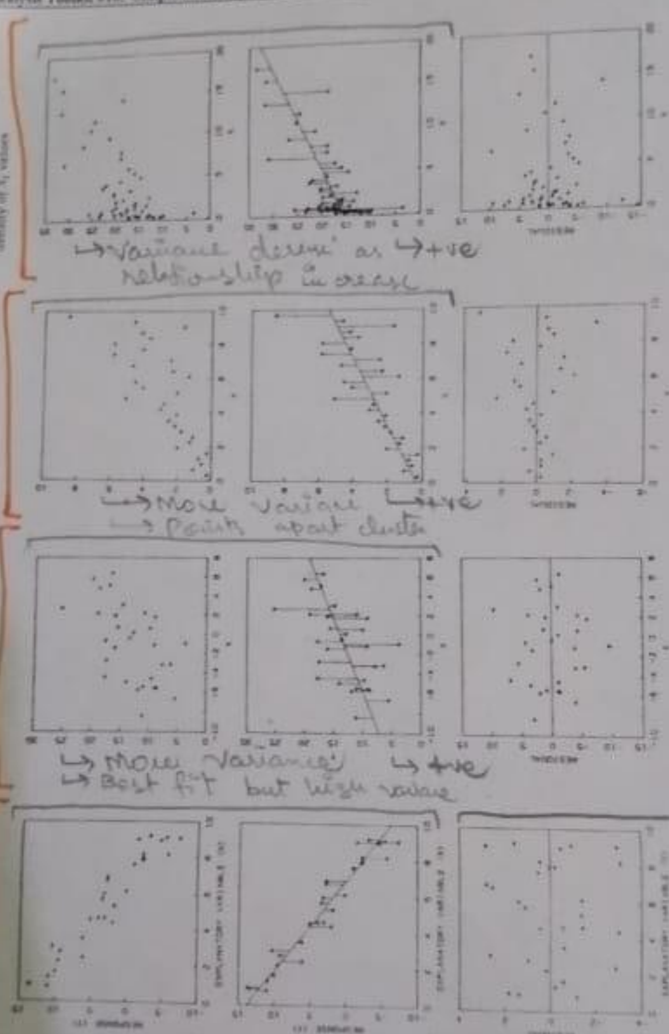
Common pitfalls in simple linear regression

- Mistakenly attributing causation. Regression *assumes* that X causes Y; it cannot *prove* that X causes Y. X and Y may be strongly correlated either because X causes Y, or because Y causes X, or because some other variable(s) causes variation in both X and Y. Which brings us to:
- Overlooking hidden variables. As mentioned above, hidden variables that are correlated with both X and Y can obscure, or even distort, the dependence of Y on X.
- Overlooking serial correlation. Strong serial correlation can cause you to seriously underestimate the uncertainties in your regression results (since successive measurements are not independent, the true number of degrees of freedom is much smaller than n suggests). In time-series data, it can also produce spurious but impressive-looking trends.
- Overlooking artifactual correlation. Whenever some part of the X-axis variable also appears on the Y-axis, there is artifactual correlation between X and Y in addition to (or in opposition to) the real correlation between X and Y.
- Note: all of pitfalls listed above can occur even though r^2 is large; indeed, all of them can sometimes serve to inflate it. This is yet another reason why r^2 should rarely be the "holy grail" of regression analysis.
- Overlooking uncertainty in X. Linear regression assumes that X is known precisely, and only Y is uncertain. If there are significant uncertainties in X, the regression slope will be lower than it would have been otherwise. The regression line will still be an unbiased estimator of the value of Y that is likely to accompany a given X measurement, but it will be a biased estimator of the Y values that would arise if X could be controlled precisely.

Q. ↳ Comment on these graphs (any one emphasis)
↳ Where is variance?

at (Y-axis) lower there is more cluster

But at 0 as a result there is less variance why relationship variance increases



↳ variance decreases as ↳ +ve relationship increase

↳ More variance ↳ +ve
↳ Points apart cluster

↳ More variance ↳ +ve
↳ Best fit but high variance

↳ Best fit ↳ -ve relationship
↳ SSR
↳ less Residual^2
↳ No outlier
↳ less variance

↳ underfit
↳ more variance
↳ no linear relationship

→ Trying to find linear relationship b/w two variables b/c we are working on two var. and a straight line

Linear relationship with one highly influential outlier that alters the fitted regression line

Data that have virtually no linear relationship between X and Y, except for a highly influential outlier

Linear relationship with outlier that has high *leverage*, but little *influence* (it could profoundly alter the regression line, but doesn't, because it is consistent with the rest of the data).

Nonlinear relationship between X and Y

References:

Chambers, J. M., W. S. Cleveland, B. Kleiner and P. A. Tukey, *Graphical Methods for Data Analysis*, 395 pp., Wadsworth & Brooks/Cole Publishing Co., 1983.

Helsel, D. R. and R. M. Hirsch, *Statistical Methods in Water Resources*, 522 pp., Elsevier, 1992.

Mosteller, F. and J. W. Tukey, *Data Analysis and Regression*, 588 pp., Addison-Wesley, 1977.

*Objective of PCA: [Principal Component Analysis]

-To project high dimensional data onto lower dimensional space.

[dimensional reduction]

→ feature Selection / feature Ranking

eg: In real life we have 2000 attributes & we have to analyse which 20 attributes among those are most important.

→ [maximum information]

Derivative:

→ Set of n points (X_i, Y_i) on a scatterplot.
→ find bestfit line $\boxed{\hat{Y}_i = a + bX_i}$
→ Such that: sum of square errors [SSE] is $\Sigma[Y_i - \hat{Y}_i]^2$ to minimize error.

∴ Name of Sum of Square error is "Q"

[n errors] ← Sum of errors

$$\Rightarrow Q = \sum_{i=1}^{n} (Y_i - \hat{Y}_i)^2 \quad \text{---(1)}$$

[put value of $\hat{Y}_i$ in eq(1) but [actual − predict]; Square → to min. value]

$$\Rightarrow Q = \sum_{i=1}^{n} (Y_i - a - bX_i)^2$$

- Then, Q will be minimized at value of a & b. Take partial derivative w.r.t a & b.

⇒ Here, take Partial Derivative w.r.t a

$$\Rightarrow \frac{\partial Q}{\partial a} = \sum_{i=1}^{n} 2.[Y_i - a - bX_i] \cdot \frac{\partial Q}{\partial a}[Y_i - a - bX_i]$$

$$= \sum_{i=1}^{n} -2[Y_i - a - bX_i]$$

$$\sum_{i=1}^{n} 2[Y_i - a - bX_i] \cdot [-(1)]$$

$$\sum_{i=1}^{n} -2[Y_i - a - bX_i]$$

Take [-] common;

$$\frac{\partial Q}{\partial a} = \sum_{i=1}^{n} 2[-Y_i + a + bX_i]$$

[-ve sign divide one rest]

$$= \sum_{i=1}^{n} 2[a + bX_i - Y_i]$$

- If we divide by 2 on both side

$$\frac{a'}{\alpha} = \sum_{i=1}^{n} \frac{\alpha[a + b\bar{x}_i - y_i]}{\alpha}$$

$$\boxed{a = \bar{y} - b\bar{x}}$$

$$\Rightarrow a' = a + bx_i - y_i$$

$$\Rightarrow \frac{\alpha \cdot a}{\alpha} = \frac{\alpha + bx_i - y_i}{\alpha}$$

$$\Rightarrow \boxed{a = y_i + bx_i}$$

$\Rightarrow$ Now take partial derivative w.r.t b.

$$\frac{\partial \theta}{\partial b} = \sum_{i=1}^{n} (y_i - a - bx_i)^2$$

$$= \sum_{i=1}^{n} 2(y_i - a - bx_i) \cdot \frac{\partial \theta}{\partial b}(y_i - a - bx_i)$$

(1) constant

$$= \sum_{i=1}^{n} 2(y_i - a - bx_i)(-x_i)$$

$$= \sum_{i=1}^{n} -2(x_i y_i - a x_i - b x_i^2) \quad \text{---(a)}$$

$\Rightarrow$ If we put value of a in to eq (a)
So, we get:

$$= \sum_{i=1}^{n} -2(x_i y_i - (\bar{y} - b\bar{x})x_i - b x_i^2)$$

Turned into $\frac{\partial \theta}{\partial b} = 0$,

$$\Rightarrow = \sum_{i=1}^{n} -2(x_i y_i - \bar{y}x_i + b\bar{x}x_i - b x_i^2) \quad \text{---(1)}$$

-2 is divide on both side:

$$= \frac{-2(x_i y_i - \bar{y}x_i + b\bar{x}x_i - b x_i^2)}{-2} = \frac{0}{-2}$$

$$(x_i y_i - \bar{y} x_i + b \bar{x} x_i - b x_i^2) = 0$$

$b \rightarrow$ constant, take out common b;
$x_i \rightarrow$ common too

$$\sum_{i=1}^{n} x_i (y_i - \bar{y}) + b(\bar{x} x_i - x_i^2) = 0$$

$$\sum_{i=1}^{n} x_i (y_i - \bar{y}) = -b(\bar{x} x_i - x_i^2)$$

$$\sum_{i=1}^{n} x_i (y_i - \bar{y}) = \sum_{i=1}^{n} b x_i (\bar{x} - x_i)$$

$$(x_i - \bar{x})$$

$$b = \frac{\sum_{i=1}^{n} (y_i - \bar{y})}{\sum_{i=1}^{n} (x_i - \bar{x})}$$

Multiply & Divide by Denominator $(x_i - \bar{x})$

$$b = \frac{\sum_{i=1}^{n} (y_i - \bar{y})}{\sum_{i=1}^{n} (x_i - \bar{x})} \times \frac{(x_i - \bar{x})}{(x_i - \bar{x})}$$

$$= \frac{\sum_{i=1}^{n} (x_i - \bar{x})(y_i - \bar{y})}{\sum_{i=1}^{n} (x_i - \bar{x})^2} = \frac{\text{Cov}(x, y)}{\text{Var}(x)}$$

a measure of relation b/w the mean value of one variable (eg. output) & corresponding values of other values (eg. time / cost)

Linear means line

Regression → Regression to the mean.
→ to return a former or less developed state

X var → independent var. [predictor var.]
Y var → dependent var. [criterion var.] [x]

Simple linear Regression plots one independent var against one dependent var [Y]

$y = mx + b$

$y' = a + bx$
$= b_0 + b_1 x$

Derivation:

① Residual / Error

$$E_i = [Y_i - (a_0 + a_1 X_i)] \quad \text{---} \quad ①$$
$$= [Y_i - a_0 - a_1 X_i]$$

Takes summation (Σ) for all the errors

$$S_r = \sum_{i=1}^{n} (E_i)^2$$

[constant of Reg. model]

$$S_r = \sum_{i=1}^{n} [Y_i - a_0 - a_1 X_i]^2$$

[sum of squares residual] [b/c of least error]

→ can't make it exact 0 b/c it mean the line go through all datalane which is not the case in regression.

- Take derivatives;

$$\frac{\partial S_r}{\partial a_0} = 0 \quad ; \quad \frac{\partial S_r}{\partial a_1} = 0$$

two eqⁿ — two unknown

$a_0 a_1 = ?$

② $$\frac{\partial S_r}{\partial a_0} = \sum_{i=1}^{n} 2\left[y_i - a_0 - a_1 x_i\right][-1] = 0$$

↳ chain rule

$$\Rightarrow -2y_i + 2a_0 + 2a_1 x_i = 0$$

const at one place

$$2a_0 + 2a_1 x_i - 2y_i = 0$$

$$2\left[a_0 + a_1 x_i - y_i\right] = 0$$

$$\sum_{i=1}^{n}\left[a_0 + a_1 x_i - y_i\right] = 0/2 = 0$$

$$\sum_{i=1}^{n} a_0 + a_1 \sum_{i=1}^{n} x_i - \sum_{i=1}^{n} y_i = 0$$

$$n a_0 + a_1 \sum_{i=1}^{n} x_i = \sum_{i=1}^{n} y_i$$

$$\frac{\partial S_r}{\partial a_1} = \sum_{i=1}^{n} 2[y_i - a_0 - a_1 x_i][-x_i] = 0$$

$$= -2\sum_{i=1}^{n} x_i y_i + 2\sum_{i=1}^{n} a_0 x_i + 2\sum_{i=1}^{n} a_1 x_i^2 = 0$$

$$\Rightarrow 2\left[a_0 \sum_{i=1}^{n} x_i + a_1 \sum_{i=1}^{n} x_i^2 - \sum_{i=1}^{n} x_i y_i\right] = 0$$

$$\Rightarrow a_0 \sum_{i=1}^{n} x_i + a_1 \sum_{i=1}^{n} x_i^2 - \sum_{i=1}^{n} x_i y_i = 0$$

$$\Rightarrow a_0 \sum_{i=1}^{n} x_i + a_1 \sum_{i=1}^{n} x_i^2 = \sum_{i=1}^{n} x_i y_i$$

matrix form

$$\begin{bmatrix} n & \sum_{i=1}^{n} x_i \\ \sum_{i=1}^{n} x_i & \sum_{i=1}^{n} x_i^2 \end{bmatrix} \begin{bmatrix} a_0 \\ a_1 \end{bmatrix} = \begin{bmatrix} \sum_{i=1}^{n} y_i \\ \sum_{i=1}^{n} x_i y_i \end{bmatrix}$$

$$a_1 = \frac{n\sum_{i=1}^{n} x_i y_i - \sum_{i=1}^{n} x_i \sum_{i=1}^{n} y_i}{n\sum_{i=1}^{n} x_i^2 - \left[\sum_{i=1}^{n} x_i\right]^2}$$

$$a_0 = \frac{\sum_{i=1}^{n} y_i}{n} - a_1 \frac{\sum_{i=1}^{n} x_i}{n}$$

$$= \bar{Y} - a_1 \bar{X}$$

$$\boxed{y = a_0 + a_1 x}$$

Week: 02

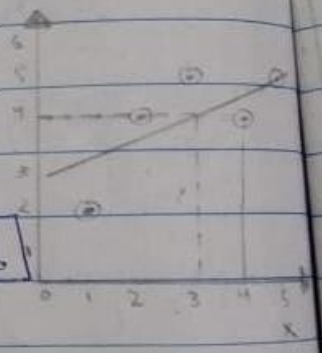
1. Examples of Linear Regression and explain derivation step by step.

2. How to plot high dimensional data? What are the techniques to plot high dimensional data?

Example

x	y	$x-\bar{x}$	$y-\bar{y}$	$(x-\bar{x})^2$	$(x-\bar{x})(y-\bar{y})$
1	2	-2	-2	4	4
2	4	-1	0	1	0
3	5	0	1	0	0
4	4	1	0	1	0
5	5	2	1	4	2
$\bar{x}=3$	$\bar{y}=4$			$\Sigma=10$	$\Sigma=6$

$$\hat{y} = b_0 + b_1 \bar{x} \quad \text{——} (1)$$



$$b_1 = \frac{\Sigma[x-\bar{x}][y-\bar{y}]}{\Sigma[x-\bar{x}]^2} \Rightarrow \frac{6}{10} \boxed{= 0.6}$$

put in (1) & $\hat{y} = 4$ (mean), $\bar{x} = 3$

Graph: It has +ve relation when X increase Y also increase.
→ It has Linear regression.

$$4 = b_0 + 0.6(3)$$

$$4 = b_0 + 1.8 \quad \text{——} (2)$$

∴ 1.8 minus with eq" (2) on both side

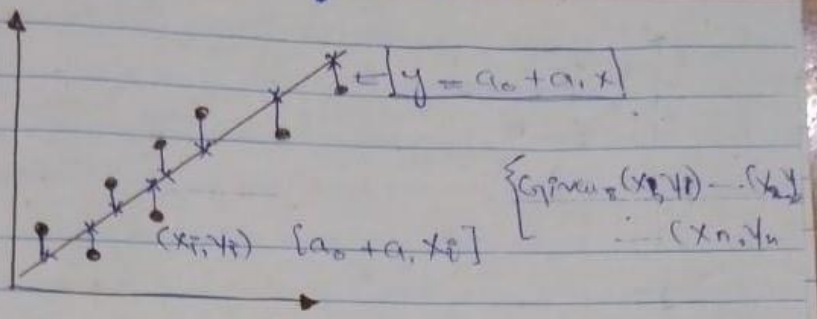
$$4 = b_0 + 1.8$$
$$-1.8 \quad -1.8$$
$$2.2 = b_0 \qquad \boxed{b_0 = 2.2}$$

$\therefore$ $b_0 = 2.2$, $b_1 = 0.6$

$$\boxed{\hat{y} = 2.2 + 0.6(\bar{x})}$$

Ans.

Linear Regression

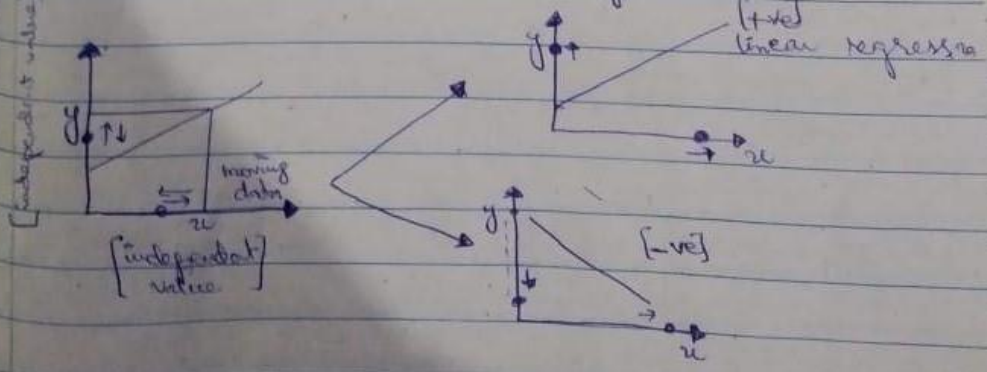


↳ trying to best fit with given data.

- best fit, straight line to
- minimize of difference b/w value of data set.

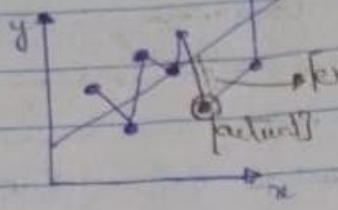
- What is straight line predict? Difference b/w ✕ what your observing & predicting.

[× ⟶ predicting
[∘ ⟶ observing



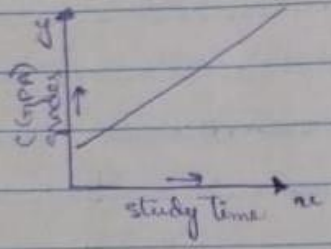
relationship
Y also
sion.

2021.03.30 16:17



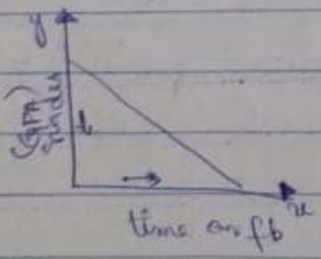
→ we have to minimize value.

eg

① 

→ study time increase Grades increase.

→ +ve relationship

② 

→ time on fb increase Grades decrease

→ -ve relationship

x → independent var
control
manipulate
change

y → dependent var

LINEAR REGRESSION

Co-ordinate

$\{(0,2), (1,3), (2,5), (3,4), (4,6)\}$

find linear regression line $y = ax + b$

Estimate value of y when $x = 10$

Sol:

x	y	xy	x^2
0	2	0	0
1	3	3	1
2	5	10	4
3	4	12	9
4	6	24	16
$\Sigma x = 10$	$\Sigma y = 20$	$\Sigma xy = 49$	$\Sigma x^2 = 30$

We know calculate a & b using linear regression

Formula:

$$a = \frac{n \sum_{i=1}^{n} X_i Y_i - \sum_{i=1}^{n} X_i \sum_{i=1}^{n} Y_i}{n \sum_{i=1}^{n} X_i^2 - \left[\sum_{i=1}^{n} X_i\right]^2}$$

$$b = \frac{1}{n}\left[\sum_{i=1}^{n} Y_i - a \sum_{i=1}^{n} X_i\right]$$

$$a = [n\Sigma xy - \Sigma x \Sigma y] \;/\; [n\Sigma x^2 - (\Sigma x)^2]$$

$$= \frac{[5*49) - 10}{[5*30) - 10]}$$

$$b = (1/n)(\Sigma y - a\Sigma x) = \frac{1}{5}[20 - (0.9 * 10)]$$

$$= 2.2$$

Now that we have linear regression line $y = 0.9x + 2.2$, substitute x_i by 10 to find the value of the corresponding y

$$y = 0.9 * 10 + 2.2 \quad [= 11.2]$$

Answer:

Techniques for Visualizing High-Dimensional Data:-

KNN-Algorithm:

" K-Nearest Neighbour or KNN algorithm is a simple algorithm which uses the entire dataset in its training phase whenever a prediction is required for an unseen data instance, it searches through the entire training dataset for k-most similar instance and the data with the most similar instance is finally returned as the prediction.

- KNN is often used in search applications. [where you're looking for similar items like find items similar to this one]
- Algorithm suggests that if you're similar to your neighbours, then you're one of them.

How does it work?

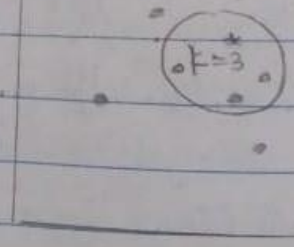
It uses very simple approach to perform classification. When test with new eg; it looks through training data & finds k-training examples that are closest to new eg. It then assigns the most common class label (among those k-training example) to the

test example.

What does 'k' in KNN algorithm represent?

↳ It represents no. of nearest neighbour points that are voting for the new test data's class.

↳ If [k=3], labels of the three closest class are checked and most common (i.e. occuring at least twice) label is assign & so on for layer ks.



Principle Component Analysis:

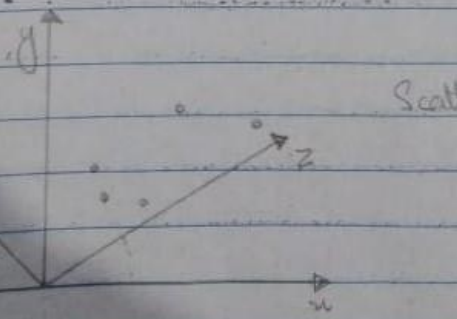
Principal component analyses is basically a statistical procedure to convert a set of observation of possibly correlated variables into a set of values of linearly uncorrelated variables.

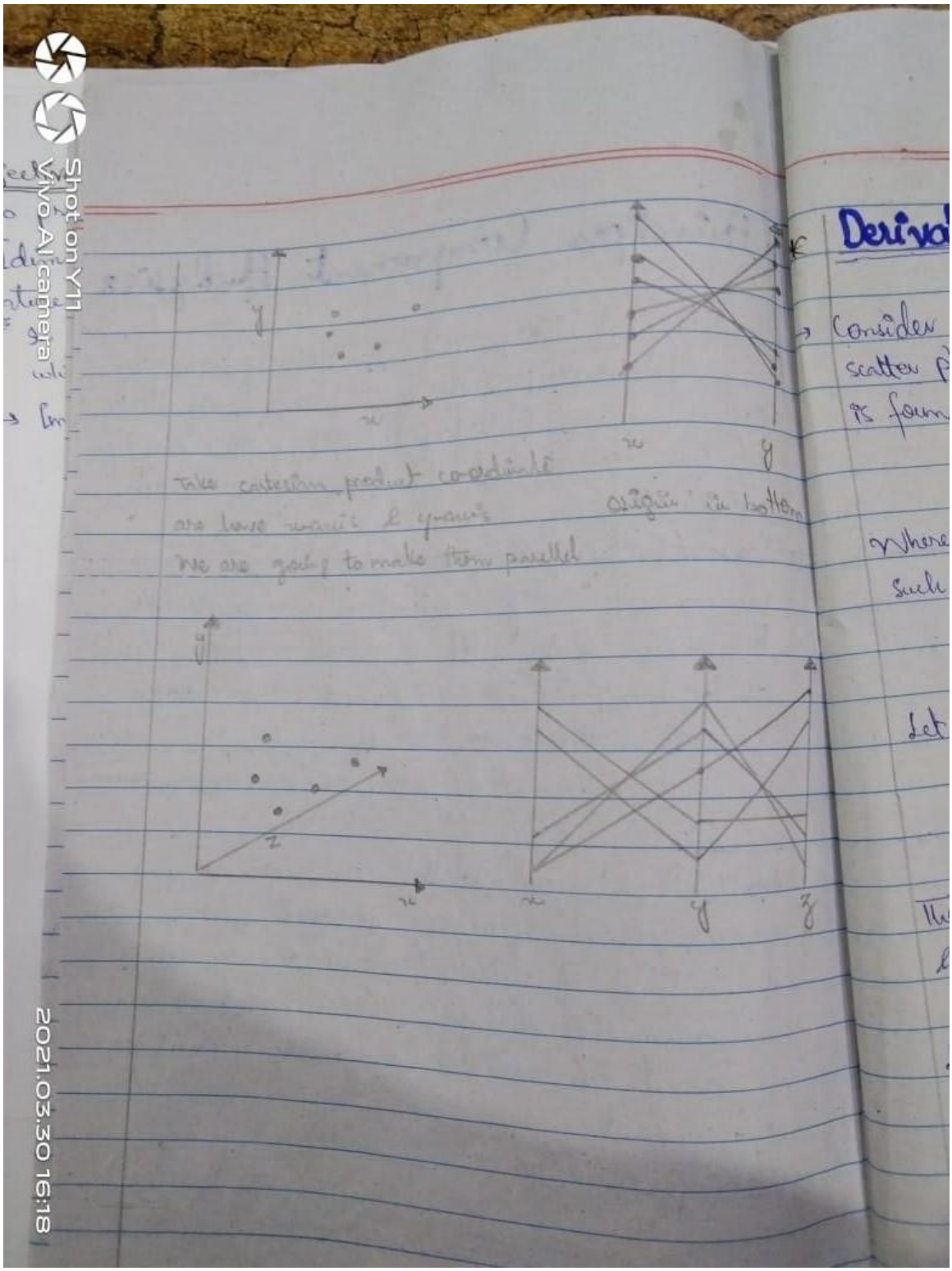
↳ Each of the principal components is chosen in such a way so that it would describe the most of the still available variance and all these principal components are orthogonal to each other. In all principal components first principal components has maximum variance

x—x

Parallel plot or parallel co-ordinates plot allows to compare the feature several individual observation (series) on a set of numeric variables. Each vertical bar represents a variable and often has its own scale.

Example:




Shot on Y11
vivo AI camera
2021.03.30 16:18
Derivat
Consider
scatter
is foun
where
such
Let
take centering point coordinate
are here x and y axis's
we are going to make them parallel
origin in bottom
x
y
z

Derivation:

Consider a set of n points (x_i, y_i) on a scatter plot. The equation of the best fit is found to be.

$$\boxed{\hat{y}_i = a + bx_i}$$

Where,

Such that the sum of squared errors in y_i :

$$\sum (y_i, \hat{y}_i)^2 \quad \text{is minimized}$$

Let,

$$Q = \sum (y_i - \hat{y}_i)^2$$

$$Q = \sum (y_i - (a + bx_i))^2$$

$$= \sum [y_i - a - bx_i]^2$$

Then, Q will be minimized at values of a & b for $\frac{\partial Q}{\partial a} = 0$ & $\frac{\partial Q}{\partial b} = 0$

first of these conditions is,

$$\frac{\partial Q}{\partial a} = \sum 2[y_i - a - bx_i](-1)$$

$$0 = -2 \sum [y_i - a - bx_i]$$

$$\sum [y_i - a - bx_i] = 0$$

$$\sum y_i - na - \sum bx_i = 0$$

$$\sum y_i - na - \sum bx_i = 0$$

$$\sum y_i - \sum bx_i = na$$

$$a = \frac{\sum y_i}{n} - \frac{\sum bx_i}{n}$$

$$\because \left[\text{Mean} = \bar{X} = \frac{\sum x_i}{n} \right]$$

$a = \bar{y} - b\bar{x}$ is y intercept.

The second condition for minimising Q is

$$\frac{\partial Q}{\partial b} = \frac{\partial}{\partial b} \sum [y_i - a - bx_i]^2$$

$$0 = \sum 2[y_i - a - bx_i][-x_i]$$

$$0 = \sum -2[x_i y_i - a x_i - b x_i^2]$$

$$\sum [x_i y_i - a x_i - b x_i^2] = 0$$

$$\sum x_i y_i - \sum a x_i - \sum b x_i^2 = 0$$

$$\because [a = \bar{y} - b\bar{x}]$$

$$\sum x_i y_i - \sum [\bar{y} - b\bar{x}] x_i - \sum b x_i^2 = 0$$

$$\sum x_i y_i - \sum \bar{y} x_i - \sum b \bar{x} x_i - \sum b x_i^2 = 0$$

$$\because \left[\frac{\sum x_i}{n} = \bar{x} \Rightarrow \sum x_i = \bar{x} n \right]$$

$$\sum x_i y_i - \sum x_i [\bar{y} + b\bar{x}] - \sum b x_i^2 = 0$$

$$\sum x_i y_i - n\bar{x}[\bar{y} + b\bar{x}] - \sum b x_i^2 = 0$$

$$\sum x_i y_i - n\bar{x}\bar{y} - n b \bar{x}^2 - \sum b x_i^2 = 0$$

$$\sum x_i y_i - n\bar{x}\bar{y} = nb\bar{x}^2 - \sum b x_i^2$$

$$\sum x_i y_i - n\bar{x}\bar{y} = b[n\bar{x}^2 - \sum x_i^2]$$

Re-write derivation above steps & give reasons.

$$b = \frac{\sum x_i y_i - n\bar{x}\bar{y}}{\sum x_i^2 - n\bar{x}^2}$$

$$b = \frac{\sum [x_i - \bar{x}][y_i - \bar{y}]}{\sum [x_i - \bar{x}]^2}$$

a = y-intercept
b = slope

given a set of 'n' points
partial derivative of [error] respect to a

$$\frac{\partial \theta}{\partial a} = \frac{\partial}{\partial a}[y_i - a - bx_i] \cdot \frac{\partial}{\partial a}[y_i - a - bx_i]$$

summation apply kia [wai]

$$= \sum 2[y_i - a - bx_i](-1)$$

$$= \sum -2[y_i - a - bx_i]$$

$$= \sum y_i - na - b\sum x_i - 0$$

$[a_1 + a_2 + a_3 \dots a_n = \sum a$

$$na = \sum y_i - b\sum x_i$$

$$a = \frac{\sum y_i}{n} - b\frac{\sum x_i}{n}$$

$$\boxed{a = \bar{y} - b\bar{x}}$$

∵ b/c X & Y are vectors/matrices thats why they're in capital.

finding b:

$$\frac{\partial}{\partial b}\left[\sum (Y_i - a - bX_i)^2\right] = 0$$

$$\sum 2[Y_i - a - bX_i]\frac{\partial}{\partial b}[Y_i - a - bX_i] = 0$$

$$\sum 2[Y_i - a - bX_i][-X_i] = 0$$

$$\sum -2X_i[Y_i - a - bX_i] = 0$$

$$\sum X_i[Y_i - a - bX_i] = 0$$

$$\sum [X_i Y_i - aX_i - bX_i^2] = 0$$

substitute the value of a i.e, $a = \bar{Y} - b\bar{X}$

$$\sum [X_i Y_i - (\bar{Y} - b\bar{X})X_i - bX_i^2] = 0$$

$$\sum (X_i Y_i - \bar{Y}X_i + bX_i\bar{X} - bX_i^2) = 0$$

$$\sum (X_i Y_i - \bar{Y}X_i) + \sum (b\bar{X}X_i - bX_i^2) = 0$$

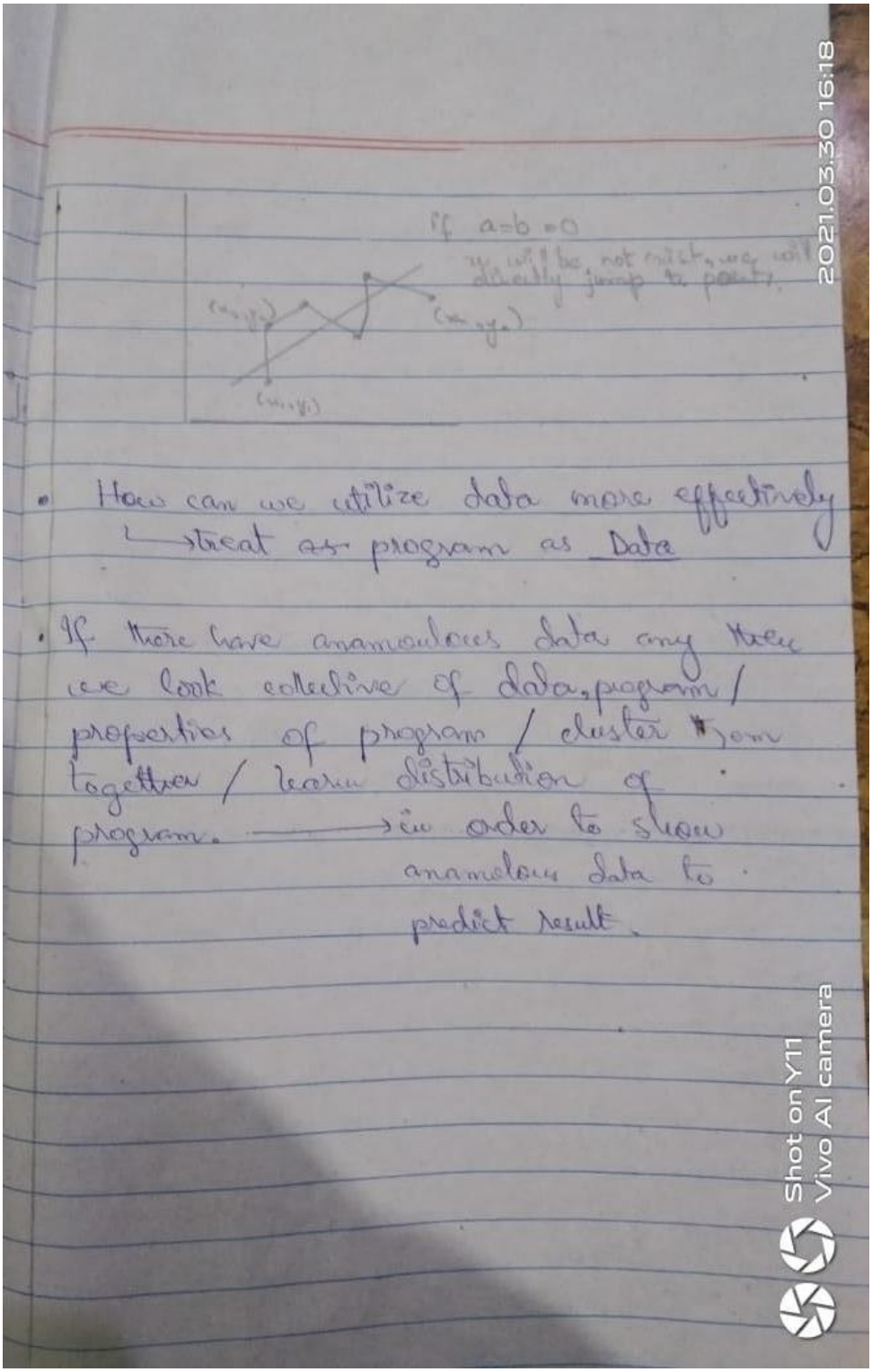
$$\sum (X_i Y_i - \bar{Y}X_i) + b\sum (\bar{X}X_i - X_i^2) = 0$$

$$\sum (X_i Y_i - \bar{Y}X_i) - b\sum (X_i^2 - \bar{X}X_i) = 0$$

$$\boxed{b = \frac{\sum [X_i Y_i - \bar{Y}X_i]}{\sum [X_i^2 - \bar{X}X_i]}}$$

a & b must not be 0, but nearest to 0 b/c if a & b is 0 then w will become 0 & graph will only play y-coordinate

• How
• If



- How can we utilize data more effectively
  → treat as program as Data

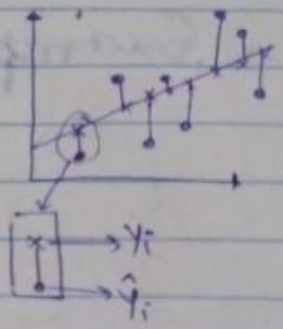
- If there have anamoulous data any then we look collective of data, program / properties of program / cluster them together / learn distribution of program. ——→ in order to show anamolous data to predict result.

* SSE ⟹ Sum of Square Errors

↳ find bestfit line. through error reduction

↳ errors ⟶ d/f b/w actual value & predicting value.

$$SSE = \sum_{i=1}^{n} [Y_i - \bar{Y}]^2$$

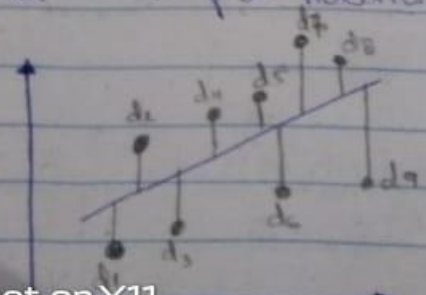


* MSE ⟹ Method of Least Square
Mean Squared Error

$$MSE = \frac{1}{n} \sum_{i=1}^{n} [Y_i - (b_0 + b_1 X_i)]^2$$

* OLS ⟹ Ordinary least Square Errors

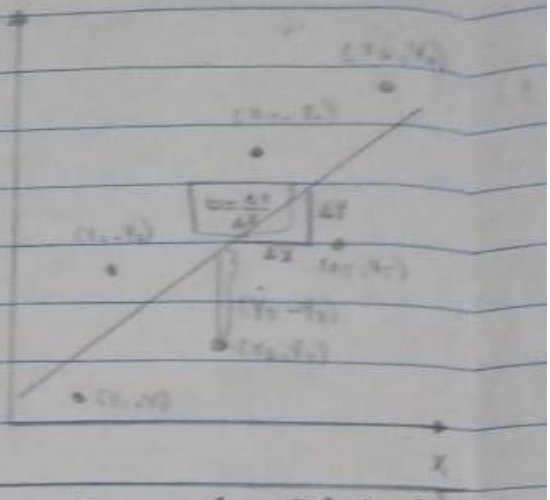
↳ Error ⟶ Simple means Standard Deviation



$$D = d_1^2 + d_2^2 + d_3^2 + d_4^2 + d_5^2 + d_6^2 + d_7^2 + d_8^2 + d_9^2$$

$$b = \frac{\sum [x-\bar{x}][y-\bar{y}]}{\sum [x-\bar{x}]^2}$$

$$a = \bar{y} - b\bar{x}$$



* Example 1:-

x	y	$x-\bar{x}$	$y-\bar{y}$	$[x-\bar{x}]^2$	$(x-\bar{x})(y-\bar{y})$
0	16.16	-125	1.03	15,625	-128.75
50	15.74	-75	0.61	5,625	-45.75
100	15.29	-25	0.16	625	-4
150	15.29	25	0.16	625	4
200	14.36	75	-0.77	5625	-57.75
250	13.99	125	-1.14	15,625	-148.75
$\bar{x}=125$	$\bar{y}=15.13$			$\Sigma = 43,750$	$\Sigma = -381$

$$\hat{y} = b_0 + b_1 \bar{x} \quad \text{---} (1)$$

$$b = \frac{\sum [x-\bar{x}][y-\bar{y}]}{\sum [x-\bar{x}]^2} = \frac{-381}{43,750} = -8.7 \times 10^{-3}$$

put in (1) & $\hat{y} = 15.13$ & $\bar{x} = 125$

$$15.13 = b_0 + (-8.7 \times 10^{-3})(125)$$

$$15.13 = b_0 - 1.088 \quad \text{---} (1)$$

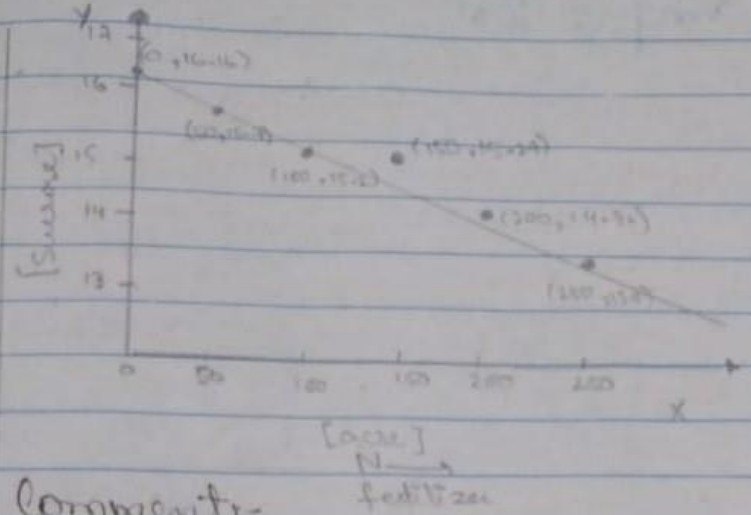
$+ 1.088$ in (1),

$$16.218 = b_0$$

$$a = 16.218$$

$$b = -8.7 \times 10^{-3}$$

$$\hat{y} = 16.218 - 8.7 \times 10^{-3} x$$



* Comments-

↳ This Graph relation shows ~~negative~~ ~~positive~~ (-ve) relationship.

↳ When Quantity of Sugar increase the N fertilizer par are also ~~increase~~ decrease

↳ There linear regression shows Best fit

↳ There is NO outliers

↳ Have less Residual / Error Square

* Example 2:

X	Y	$x-\bar{X}$	$Y-\bar{Y}$	$(x-\bar{X})^2$	$(x-\bar{X})(Y-\bar{Y})$
16.4	268.96	−1.25	265.68	1.5625	−332.1
17.2	295.84	−0.45	292.56	0.2025	−131.652
17.6	309.76	−0.05	306.48	2.5×10^{-3}	−15.324
18.0	324.00	0.35	320.72	0.1225	112.252
18.2	331.24	0.55	327.96	0.3025	180.378
18.5	342.25	0.85	338.97	0.7225	288.1245
$\bar{X}=17.65$	$\bar{Y}=3.28$			$\Sigma=2.915$	$\Sigma=101.6785$

$$\hat{y} = b_0 + b_1 \bar{X} \quad \text{——} (1)$$

$$b_1 = \frac{\Sigma[x-\bar{x}][y-\bar{y}]}{\Sigma[x-\bar{x}]^2} = \frac{101.6785}{2.915} \quad [= 34.881]$$

put in (1) & $\bar{y} = 3.28$, $\bar{X} = 17.65$

$$3.28 = b_0 + 34.881(17.65)$$

$$3.28 = b_0 + 615.64 \quad \text{——} (11)$$

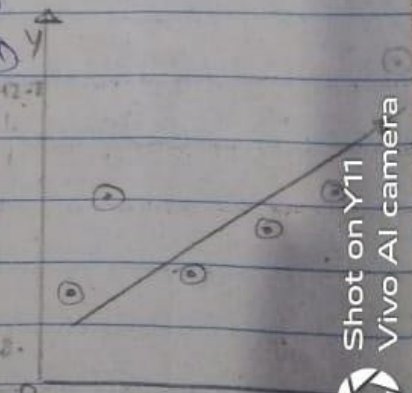
$\therefore -615.64$ on both sides; (11)

$$[-612.36 = b_0]$$

$b_1 = 34.881$

$b_0 = -612.36$

$\hat{Y} = -612.36 + 34.881(\bar{X})$



Comments

↳ there is [+ve] positive relationship

↳ If mice body incrs, its liver wt. [mice body] increase too.

WEEK: 03

- **INDIVIDUAL ASSIGNMENT:**

  Draw a scatter plot with 100 dummy points and mean zero.

- **GROUP ASSIGNMENTS:**

1. Find SE relevant dataset so that we can start applying AI algorithms for:
   a) Visualization
   b) Effort estimation
   c) Cost estimation
   d) Task scheduling
   e) Risk estimation
   f) Quality assurance

2. Design a mockup/ form to collect Projects data from classmates and UBIT Alumni.

3. Prepare a list of standard repositories and tabulate it.

Task:

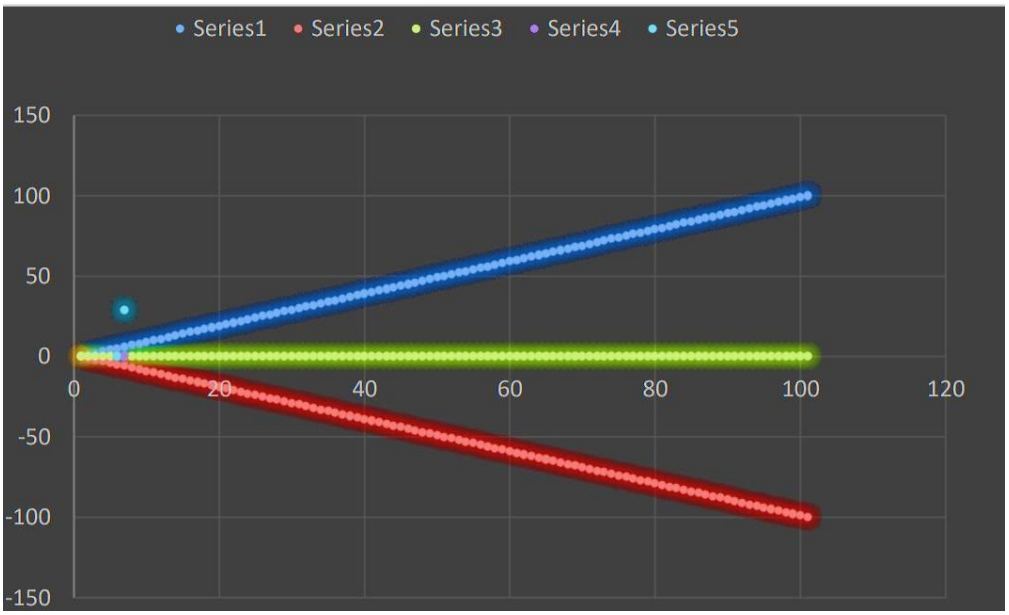
Draw a scatter plot with 100 dummy points and mean zero.

X	Y	MEAN	X	Y	MEAN
1	-1	0	40	-40	0
2	-2	0	41	-41	0
3	-3	0	42	-42	0
4	-4	0	43	-43	0
5	-5	0	44	-44	0
6	-6	0	45	-45	0
7	-7	0	46	-46	0
8	-8	0	47	-47	0
9	-9	0	48	-48	0
10	-10	0	49	-49	0
11	-11	0	50	-50	0
12	-12	0	51	-51	0
13	-13	0	52	-52	0
14	-14	0	53	-53	0
15	-15	0	54	-54	0
16	-16	0	55	-55	0
17	-17	0	56	-56	0
18	-18	0	57	-57	0
19	-19	0	58	-58	0
20	-20	0	59	-59	0
21	-21	0	60	-60	0
22	-22	0	61	-61	0
23	-23	0	62	-62	0
24	-24	0	63	-63	0
25	-25	0	64	-64	0
26	-26	0	65	-65	0
27	-27	0	66	-66	0
28	-28	0	67	-67	0
29	-29	0	68	-68	0
30	-30	0	69	-69	0
31	-31	0	70	-70	0
32	-32	0	71	-71	0
33	-33	0	72	-72	0
34	-34	0	73	-73	0
35	-35	0	74	-74	0
36	-36	0	75	-75	0
37	-37	0	76	-76	0
38	-38	0	77	-77	0

X	Y	MEAN	X	Y	MEAN
39	**-39**	**0**	**78**	**-78**	**0**
79	**-79**	**0**	**89**	**-89**	**0**
80	**-80**	**0**	**90**	**-90**	**0**
81	**-81**	**0**	**91**	**-91**	**0**
82	**-82**	**0**	**92**	**-92**	**0**
83	**-83**	**0**	**93**	**-93**	**0**
84	**-84**	**0**	**94**	**-94**	**0**
85	**-85**	**0**	**97**	**-97**	**0**
86	**-86**	**0**	**98**	**-98**	**0**
87	**-87**	**0**	**99**	**-99**	**0**
88	**-88**	**0**	**100**	**-100**	**0**

Mean = 0

Standard deviation = 29.01149



Group work

Task: 01

1. Find SE relevant dataset so that we can start applying AI algorithms for:
    a) Visualization
    b) Effort estimation
    c) Cost estimation
    d) Task scheduling
    e) Quality assurance
    f) Risk estimation
    g) Security
    h) user experience
    i) performance

Dataset Link:

https://www.kaggle.com/semustafacevik/software-defect-prediction-data-analysis

Task: 02

Design a mockup/ form to collect Projects data from classmates and UBIT Alumni.

https://docs.google.com/forms/d/e/1FAIpQLScjdq8UQtUqdMksmkkqkGdIlyqnu4ExA6ObShpUZJjgube1qw/viewform?usp=sf_link

	A	B	C	D	E	F
1	Timestamp	Your Name:	Email:	Attribute#1	Attribute#2	Attribute#3
2	3/31/2021 0:33:30	Wajiha Hanif	b18158064.wajihahanif@gmail.com	Security	Quailty	Risk estimation
3	3/31/2021 0:35:48	Wajiha Hanif	b18158064.wajihahanif@gmail.com	visualization	Robustness	Efficiency
4	3/31/2021 0:37:23	Wajiha Hanif	b18158064.wajihahanif@gmail.com	Scalabilty	Functionality	cost estimate
5	3/31/2021 0:37:34	Syeda Musfira Masroor	masroormusfira@gmail.com	Maintainability	Dependability	Efficiency
6	3/31/2021 0:40:22	Syeda Musfira Masroor	masroormusfira@gmail.com	Reliability	Usability	Extensibility
7	3/31/2021 0:48:41	Houra Batool	hourabatool@gmail.com	Reliability	Correctness	Portability
8						
9						
10						

Task: 04

Prepare a list of standard repositories and tabulate as follows:

SOURCE LINK ID	WEB LINK OR ARITHMETIC	REFERENCE PAPER	GITHUB LINK
1)	Kaggle environments	IDD: A Dataset for Exploring Problems of Autonomous Navigation in Unconstrained Environments Girish Varma, Anbumani Subramanian, Anoop Namboodiri, Manmohan Chandraker, C V Jawahar	https://github.com/Kaggle/kaggleenvironments
2)	Awesome AI Guidelines	-	https://github.com/EthicalML/awesome-artificial-intelligence-guidelines
3)	Tools and tests used in Kaggle Learn exercises	An Introduction to Machine Learning Theory and Its Applications: A Visual Tutorial with Examples by NICK MCCEA	https://github.com/Kaggle/learntools
4)	Kaggle extension for JupyterLab	-	https://github.com/Kaggle/upyterlab
5)	A dockerfile to install all of CRAN	An Introduction to Rocker: Docker Containers for R by Carl Boettiger, Dirk Eddelbuettel	https://github.com/Kaggle/docker-rcran

WEEK: 04

1. Individual Weekly Assignment of preparing a report on the significance and weakness of the study w.r.t software engineering.

2. Group weekly Assignment of visualizing the chosen dataset and comments on how effective the dataset is w.r.t software engineering.

ARTIFICIAL INTELLIGENCE IN SOFTWARE ENGINEERING

Group Assignment

Dataset: "Predictive models of Software Engineering from PROMISE Datasets"

https://www.kaggle.com/semustafacevik/software-defect-prediction-data-analysis

Group members:

Wajiha Hanif (B18158064)

Musfira Masroor (B18158057)

Houra Batool (B18158016)

Ahtisham-ud-Din (B18158001)

Predictive models of Software Engineering from PROMISE Datasets

Predictive models are one of the most important techniques that are widely applied in many areas of software engineering. There have been a large number of primary studies that apply predictive models and that present well-preformed studies and well-designed works in various research domains, including software requirements, software design and development, testing and debugging and software maintenance.

1 Predictive Models of Software Engineering

A key technology, the predictive model, has been developed to solve a range of software engineering problems over several decades. The use of predictive models is in fact becoming increasingly popular in a wide range of software engineering research areas. Predictive models are built based on different types of datasets – such as software requirements, APIs, bug reports, source code and run-time data – and provide a final output according to distinct features found in the data.

There are various predictive models commonly used in software engineering tasks that contribute to improving the efficiency of development processes and software quality. Common ones include defect prediction, API issue classification, and code smell detection.

2 Dataset

In our dataset, the available attributes are all single valued static variables. The data set includes 22 attributes (5 different lines of code measure, 3 McCabe metrics, 4 base Halstead measures, 8 derived Halstead measures, a branch-count, and 1 goal field).

1. loc : numeric % McCabe's line count of code
2. v(g) : numeric % McCabe "cyclomatic complexity"
3. ev(g) : numeric % McCabe "essential complexity"
4. iv(g) : numeric % McCabe "design complexity"
5. n : numeric % Halstead total operators + operands

6. v : numeric % Halstead "volume"

7. l : numeric % Halstead "program length"

8. d : numeric % Halstead "difficulty"

9. i : numeric % Halstead "intelligence"

10. e : numeric % Halstead "effort"

11. b : numeric % Halstead

12. t : numeric % Halstead's time estimator

13. lOCode : numeric % Halstead's line count

14. lOComment : numeric % Halstead's count of lines of comments

15. lOBlank : numeric % Halstead's count of blank lines

16. lOCodeAndComment: numeric

17. uniq_Op : numeric % unique operators

18. uniq_Opnd : numeric % unique operands

19. total_Op : numeric % total operators

20. total_Opnd : numeric % total operands

21: branchCount : numeric % of the flow graph

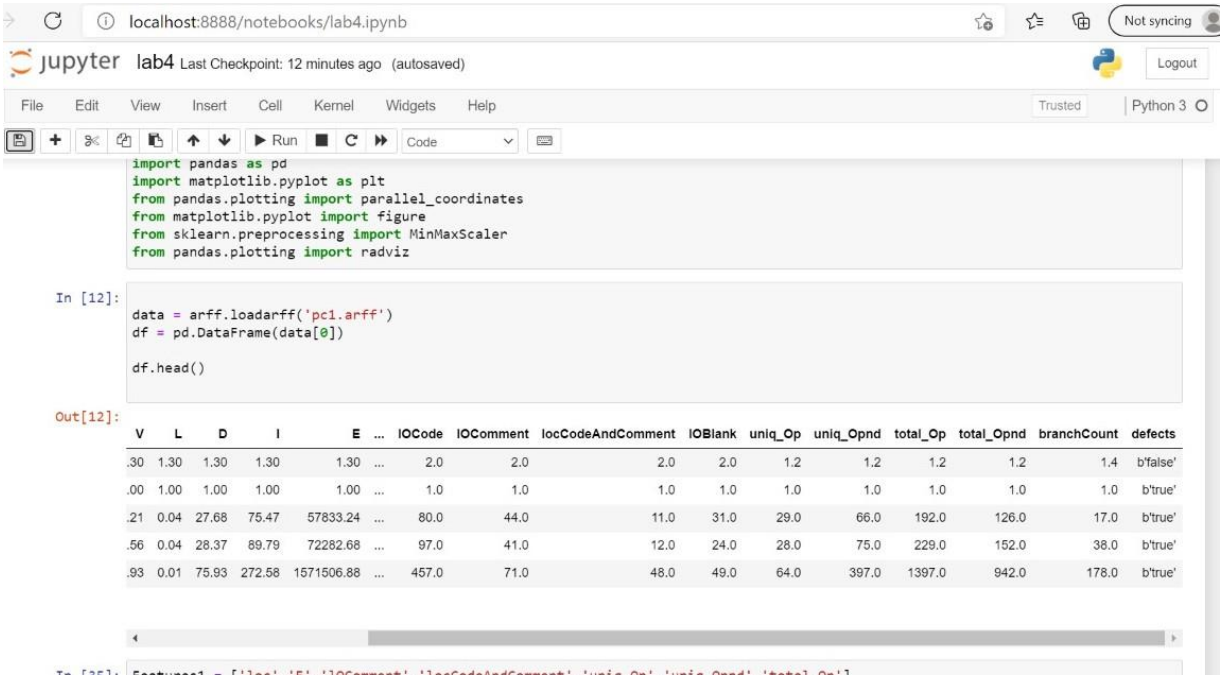
22. defects : {false, true} % module has/has not one or more

reported defects

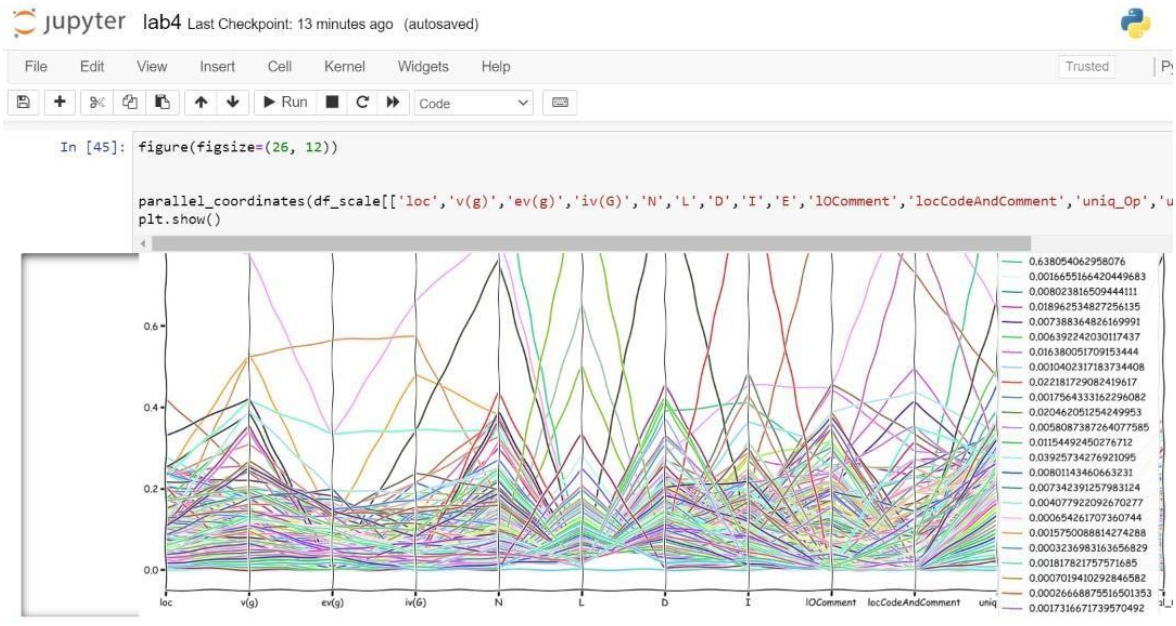
- This data came from McCabe and Halstead features extractors of source code. The McCabe and Halstead measures are "module"-based where a "module" is the smallest unit of functionality. In C or Smalltalk, "modules" would be called "function" or "method" respectively.

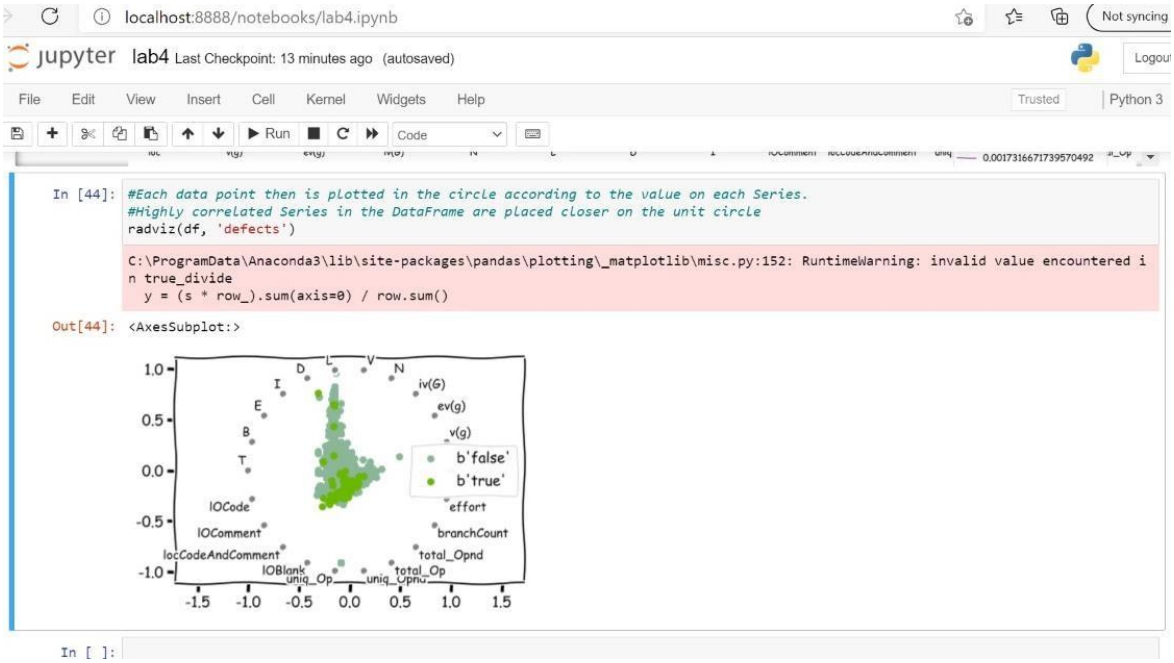
- Both McCabe and Halstead used different strategies to work with the dataset. McCabe's metrics depicts the pathway in a code module whereas Halstead estimates reading complexity by counting the number of concepts in a module; e.g. number of unique operators.

3 DATA VISUALIZATION



V	L	D	I	E	...	lOCode	lOComment	locCodeAndComment	lOBlank	uniq_Op	uniq_Opnd	total_Op	total_Opnd	branchCount	defects
.30	1.30	1.30	1.30	1.30	...	2.0	2.0	2.0	2.0	1.2	1.2	1.2	1.2	1.4	b'false'
.00	1.00	1.00	1.00	1.00	...	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	b'true'
.21	0.04	27.68	75.47	57833.24	...	80.0	44.0	11.0	31.0	29.0	66.0	192.0	126.0	17.0	b'true'
.56	0.04	28.37	89.79	72282.68	...	97.0	41.0	12.0	24.0	28.0	75.0	229.0	152.0	38.0	b'true'
.93	0.01	75.93	272.58	1571506.88	...	457.0	71.0	48.0	49.0	64.0	397.0	1397.0	942.0	178.0	b'true'





4 CONCLUSION

Results of the dataset show that the quality attribute language is the most statistically significant when calculating software cost. Moreover, if all quality requirements attributes are eliminated in the training stage and software cost is predicted based on software size only, the value of the error (MMRE) is doubled.

Linear Programming as a Baseline for Software Effort Estimation

Individual Report

- **SIGNIFICANCE:**

1) Both the methods ATLM & LP4EE are extensively use & trustworthy in the sense of its consistent data result.
2) LP4EE is more robust than ATLM (about 17%).
3) LP4EE, reduce data destabilization.
4) If we have more history than it's directly proportional to the good and reliable data.
5) This study highlights the importance of benchmarking methods in the context of software effort estimation.
6) This paper proposed a novel method based on Linear Programming (LP4EE, Linear programming for effort estimation) 7) Reference implementation of LP4EE for R environment is freely available for statistical computing.

- **WEAKNESS:**

1) This study highlights only the significance of benchmarking method in software effort estimation, but we know that benchmarking requires a lot of expertise and a vast collection of data.
2) Every organization can't execute its strategies in desired manner because of lack of information, increase dependency, lack of understanding, incorrect comparison, costly affair, and copying others.
3) Thirdly, linear programming has its own limitations.
  • Difficult to determine objective function in LPP (linear programming problem)
  • Difficult to specify the constraints even after the selection of objective function.
  • There is a possibility that the objective function and constraints may or may not be directly defined by linear in the
    equality of equations.

Week: 05

(Quiz done in class)

- Effort Estimation:
    1. Basic CoCoMo
    2. Intermediate CoCoMo
    3. Functional Points (FP)

Week: 06

- We had to mail our selected dataset

Week: 07

- Mid term

Course Supervisor : Dr. Humera Tariq

Subject: Artificial Intelligence in SE

Submission email : humera.tariq.dcs.uok@gmail.com

Submission Requirements : Handwritten Assignment + code + read me

Week 07 Lab Midterm

Step I: Download and Study about following Data set .

GitHub - notpeter/crunchbase-data: 2015 CrunchBase Data Export as CSV

Step II: Loading Data Set

Load data on any programming platform of your choice for visualization and study. After successful loading of dataset **fill** in following lines: about number of observations and number of variables.

No. of Observations = 2213 .

NO. of Variable = 18 .

Take a meaningful snapshot to show successful loading of your data. Please do not share your solution with your colleagues/class fellows.

acquisitions.csv

```
In [2]: df_acquisitions_pd = pd.read_csv(r'D:\code\Data Analysis Kaggle\05_crunchData_set\acquisitions.csv')
        df_acquisitions_pd.head(3)
```

Out[2]:

	company_permalink	company_name	company_category_list	company_country_code	company_state_code	company_region	company_city	acquirer_permalink	acquirer_name	acquirer_categor
0	/organization/003-ru	003.RU	Consumer Electronics|Electronics|Internet	RUS	48	Moscow	Moscow	/organization/media-sturn	Media Sturn	Enterprise Software|Media|Sale Mark
1	/organization/0958572-b-c-ltd	0958572 B.C Ltd	NaN	NaN	NaN	NaN	NaN	/organization/atlas-intellectual-property-mana...	ATLAS Intellectual Property Management Co.	Finance|FinTech|Mobile|Telecommunic
2	/organization/1-800-communications	1-800 Communications	NaN	USA	NY	Long Island	Hicksville	/organization/carsdirect-com	CarsDirect.com	E-Comm

After Careful study of dataset Fill Table No. 1 and prepare yourself to talk about required pre-processing techniques. Table 1 is on Page 3. It is empty, you need to fill it appropriately using your own handwriting.

Step III: Which task you prefer to perform this dataset:

State the difference between Regression and Classification and discuss that how the given dataset can be used in context of both scenarios.

Regression Scenario Explanation

- In Acquisition.csv: In acquisition price Vs time stamp: It simply means when time increase the price of good also increase and vice versa.
- In Addition.csv: Time stamp Vs value: means when the value of certain stuff increase or decrease the time automatically impact on it.

Classification Scenario Explanation

- In Acquisition: country code Vs price amount Simply it indicates the relation between the price and country of region where the goods are selling.

Table 1 Raw Variables to be filled in your own handwriting. You may add rows as per your understanding and requirement

Variable Name	Variable Description	Suggest any required pre-processing for e.g. missing data, excluded, duplicates, deleted, outliers, approx. zero variance with justification in this column
company_permalink	Link of the company	Replace nan value with whole column median
company_name	Name of the company	
company_category_list	What the actually the company is selling/manufacturing	
company_country_code	In every country has there own postal code like mobile code just like that at the one country there are various other companies existing too so government a lot each company it's code number.	Hot and codding, dummy value.
company_state_code	The above example with just the code with state wise	
company_region	Where the company is allocated and in which region in the world	NAN values replace by most common value Hot and code
company_city	Where the company is located in country, it's city name	NAN values replace by most common value Hot and code
acquirer_permalink	The acquirer of the company good/to selling one	
acquirer_name	Name of that seller	
acquirer_category_list	Categorical distribution	NAN values replace by most common value Hot and code
acquirer_country_code	It's code too, country.	
acquirer_state_code	State code of the demander	
acquirer_region	The region should also mention	NAN values replace by most common value, Hot and code
acquirer_city	City of the supplier	NAN values replace by most common value Hot and code

acquired_at	When, the date of shipping	Make its time stamp
acquired_month	month	
price_amount	Price of that good	NAN values replace by most common value
price_currency_code	The currency code, helps to change convert the currency value	

Step IV: Prepare a list of Continuous and discrete variables.

Discrete Variable	Continuous Variable
company_name	acquired_at
company_category_list	acquired_month
company_country_code	price_amount
company_state_code	price_currency_code
company_region	
company_city	
acquirer_name	
acquirer_category_list	
acquirer_country_code	
acquirer_state_code	
acquirer_region	
acquirer_city	

Step V: Prepare a list of your response and predictor variables. Will you consider all variable or able to reject some for any reason? Write Justification also.

Predictor variables	Response Variable
time	amount
Region	
Country	
State	

Step VI: Carefully read give Problem statements:

(i) To investigate the difference between successful and Un-Successful companies.

(ii) What features are required by a company to be acquired for funding?

(iii) How to predict that a company will be acquired or merged?

Add at least 2 more meaningful problem statements:

(iv) _What is the probability of the company that exist in which country will get benefits as compared?

(v) Company in what country will get high benefit but with thread of other negligence.

Step VII: Analyze data using exploratory data analysis techniques and submit your notebook/code along with your name and seat number at mentioned email by 5:00 pm today.

Step IX: Apply Regression to solve any problem of your choice with given dataset and submit your notebook/code along with your name and seat number at mentioned email by 6:00 pm today.

Step X: Write 5 projects here as discussed in class before 2 weeks. Ask your CR, if you were absent.

Week: 08

- Help for midterm.

Week: 09

- Regression Statistics Table

20

Todays Homework: Fill in the Table

Regression Output	Result	Explanation
Multiple R	?? 0.8599	R = square root of R^2
R Square	?? 0.7395	R^2
Adjusted R Square	?? −7.5176	Adjusted R^2 used if more than one x variable
Standard Error	?? 20.309	This is the sample estimate of the standard deviation of the error
Observations	8	Number of observations used in the regression (n)

Week : 09 Task

	Regression Output	Result	Explanation
①	Multiple R	??	R = Square root of R^2
②	R square	??	R^2
③	Adjusted R^2	??	Adjusted R^2 used if more than one x variable.
④	Standard Error	??	This sample estimate of the standard deviation of the error.
-	∅		
⑤	Observation	8	No. of observation use in regression (n)

Formulase

* $R^2 = \frac{S^2_{xy}}{SS_x \times SS_y}$; Multiple $R = \sqrt{R^2}$ [correlation $y \& \bar{y}$]

* Adjusted $R^2 = 1 - \left[\frac{(1-R)^2(n-1)}{n-k-1}\right]$

* Standard error = $SE = \sigma = \frac{\sqrt{(1-R^2) * SS_y}}{\sqrt{N-2}}$

Observations:

LOC	420	380	350	400	440	380	450	420
Rating	5.5	6.0	6.5	6.0	5.0	6.5	4.5	5.0

* Multiple R^2 = ?

we need S^2_{xy}, SS_x and SS_y ?

y	x	$(\bar{x}-x_i)$	$(\bar{y}-y_i)$	$(\bar{x}-x_i)^2$	$(\bar{y}-y_i)^2$	$(\bar{x}-x_i)(\bar{y}-y_i)$
420	5.5	-0.125	15	0.0156	225	-1.875
380	6.0	0.375	-25	0.1406	625	-9.375
350	6.5	0.875	-55	0.7656	3025	-48.125
400	6.0	0.375	-5	0.1406	25	-1.875
420	5.0	-0.625	35	0.3906	1024	-21.875
380	6.5	0.875	-25	0.7656	625	-21.875
380	4.5	-1.125	45	1.2656	2025	-50.625
420	5.0	-0.625	15	0.3906	225	-9.375
3240	45			SS_x =	SS_y =	S_p = -165
$\bar{y}$ = 405	$\bar{x}$ =			3.8748	7799	
	5.625					

* $R^2 = \frac{S^2_{xy}}{SS_x \times SS_y} = \frac{-165^2}{(3.8748)(7799)} \Rightarrow 0.9008$

* R square = $R = \sqrt{0.9008} \Rightarrow 0.9491$

* Standard Error =

$$S_t = \sqrt{\frac{(1-R)^2 (n-1) SS_y}{\sqrt{N-2}}} \Rightarrow \sqrt{\frac{(1-0.9491)^2 * 7799}{\sqrt{8-2}}}$$

$$= \frac{27.8147}{2.4494}$$

$$= 11.3557$$

* Adjusted R^2

$$\Rightarrow 1 - \left[\frac{(1-R)^2 (n-1)}{n-k-1}\right]$$

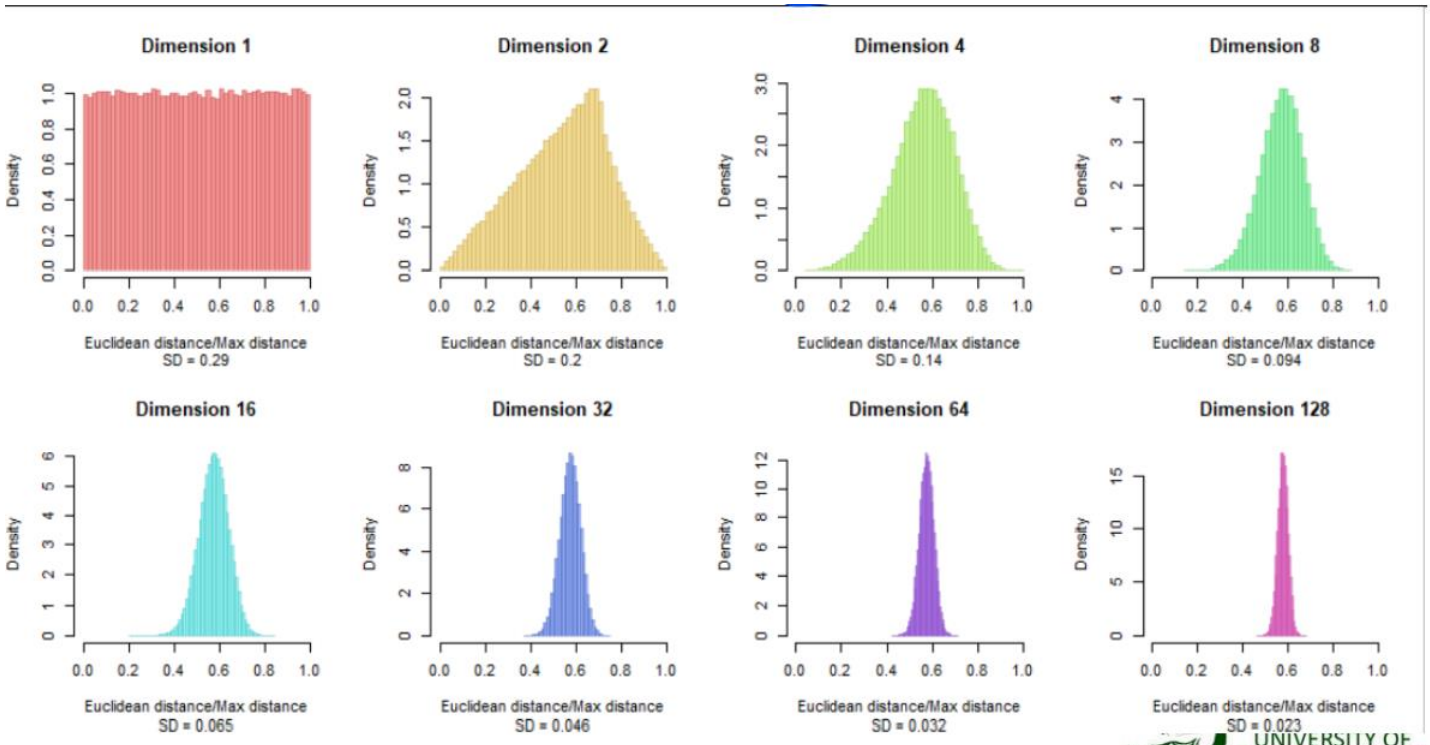
→ Null b/c of independent var.

Week: 10

ANOVA Table

- Giving 5 pictures based on artificial intelligence concepts. Take Print of each picture, Explore and write at-least 15 technical/AI relevant points that shows your understanding.

1) Density plots and Dimension curse



What is Curse of Dimensionality in Machine Learning? (mygreatlearning.com)

UNIVERSITY OF KARACHI

Week: 10

Density Plots & Dimension Curses

1. In ML, we often have high dimensional data. If we're recording 60 different metrics for each of our shoppers, we're working in a space with 60 dimensions.
2. If were analyzing image vectors grayscale sized 50×50 we're working in space with 2,500 dimension.
3. If the same image is of RGB-colored, the dimensionality increases to 7,500 dimensions (one dimension for each colour channel in each pixel in the image.
4. It is refers to set of problems that arise when working with high dimensional data.
5. A dataset with large to no. of attributes generally of the order of hundred or more, refered to high dimensional data.
6. The difficulties related to training machine learning model due to high dimensional data is refered to as "Curse of Dimensionality"
7. For eg: if we trying to predict a target that is depend on two variables age & gender. If this data use to train a model that is capable of learning the mapping b/w attribute value & target, it [illegible] be generalized.

8. As long as feature unseen data comes from the distribution (a combination of values), the model could predict target accurately.

Age
- Children (0-14)
- Youth (15-24)
- Adult (25-60)
- Senior (61-above)

Gender:
- Male
- Female

Age Group	Gender	Target
Children	Male	T1
Youth	Male	T2
Adult	Male	T3
Senior	Male	T4
Children	Female	T5
Youth	Female	T6
Adult	Female	T7
Senior	Female	T8

9. A density plot of distances b/w points and probability of frequency of occurence of distance is created for d/f dimensions.

10. For one-dimensional cases, we see that density is approximately uniform.

11. As the no. of dimensions increase, we see

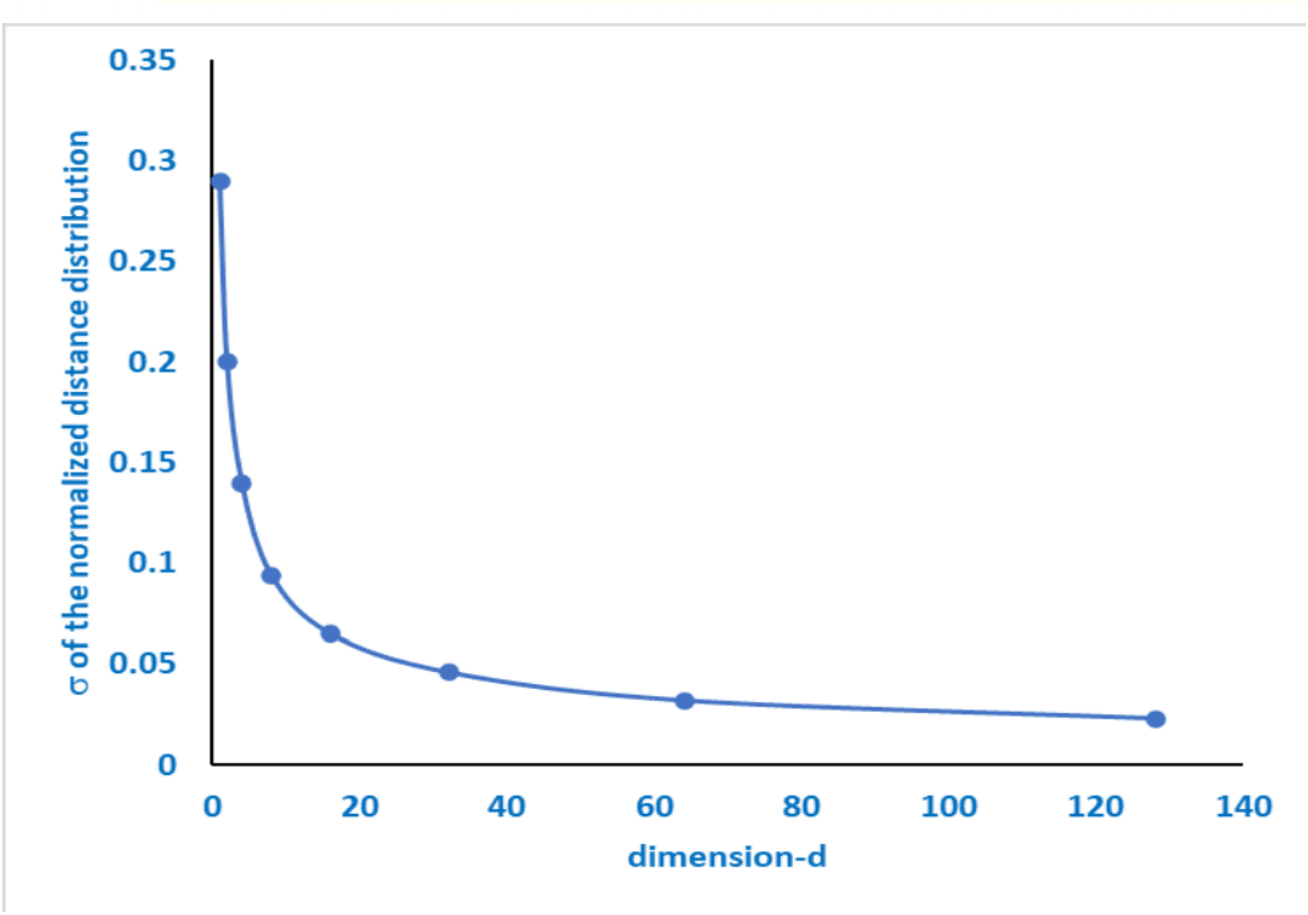
the spread of frequency plot decreases, i.e. indicating distance b/w d/f samples or points tend towards a single value as dimension increase.

12c To mitigate problems associated with high dimensional data a suite of techniques generally reffered to Dimensionality reduction technique are used.

13c Feature selection techniques:
the attributes are tested for their worthiness and then selected or eliminated.
↳ Low variance filter
↳ High correlation filter
↳ Multi collinearity
↳ Feature Ranking
↳ Forward Selection

14c Feature Extraction techniques:
↳ PCA [Principal component Analysis]
↳ Factor Analysis [FA]
↳ ICA [Independent Component Analysis]

2) Standard Deviation and Dimension Curse

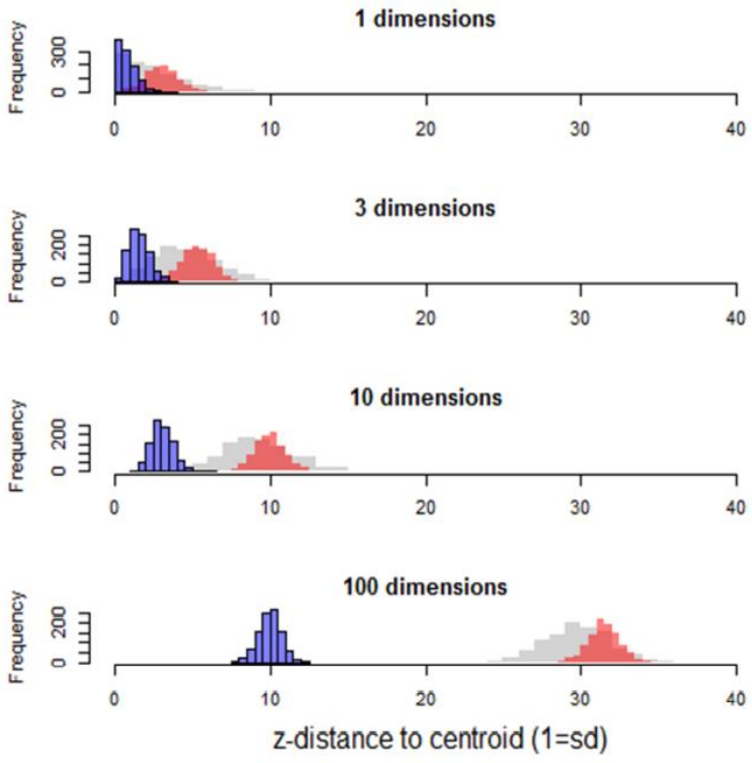


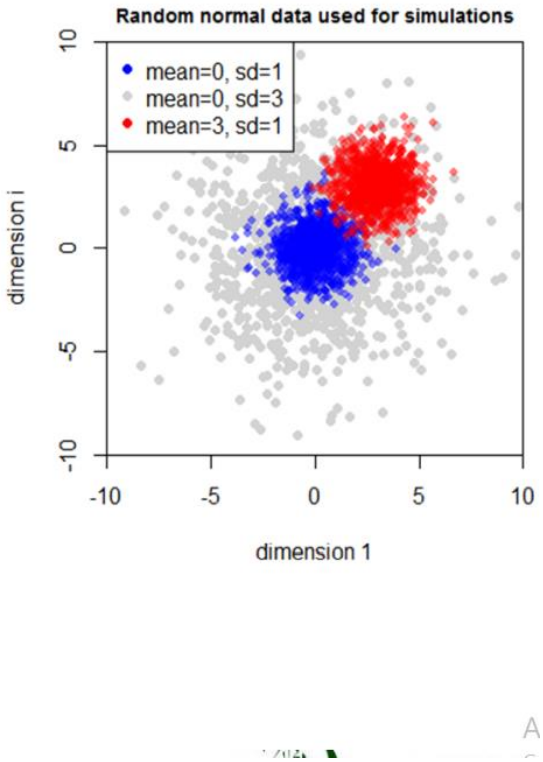
Standard Deviation & Dimension Curse:

- To measure how spreadout the numbers are in the dataset, we use standard deviation.
- If the elements in the dataset are spread further apart from their mean value, then they'll have significantly larger values of standard deviation.
- Similarly if the elements don't deviate significantly from the mean value, then they will have smaller standard deviation.
- The graph also shows that standard deviation inversely proportional to dimension. i.e
- Hughes Phenomenon shows that as the no. of features increase, the classifier performance increase as well until we reach the optimal no. of features. Adding new features based on the same size as the training set will then degrade the classifiers performance.
- Curse of dimensionality in d/s function:-

$$d(p,q) = \sqrt{\sum_{i=1}^{n} (p_i - q_i)^2}$$

3) Higher dimension is Good/bad for Prediction??



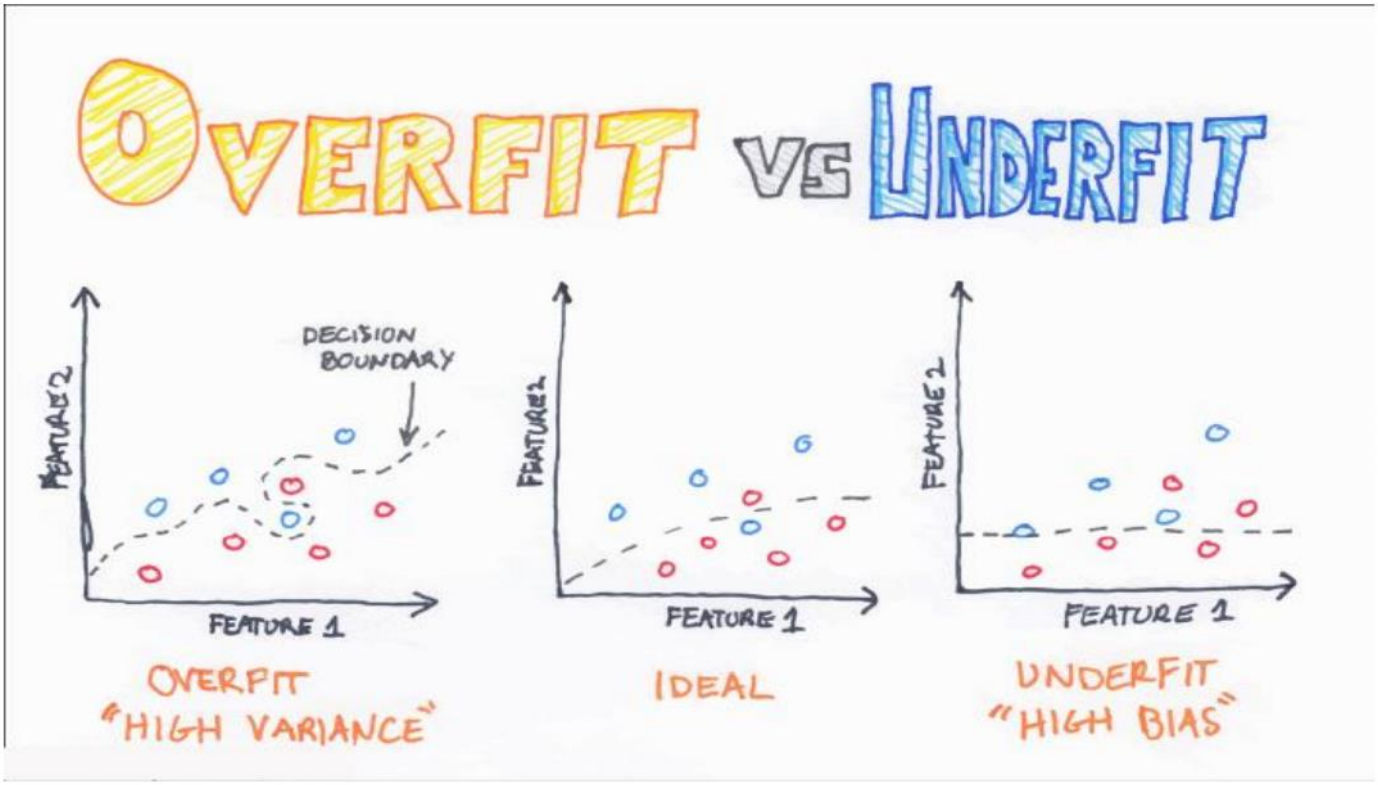


High Dimension is Good or Bad for Predictions

Many different issues arises due to large number of dimensions. Most features are observation curves in the model and also effect performances badly. The harder to cluster on first scales, dimension & frequency closed to each other. Dimension starts to create distance for frequency on second scale. More distance is created on third scales due to increase in direction and keep increase in distance, with the increase in dimension.

- High dimensional data is reduced to lower dimension that is 2D or 3D so that they can be easily plotted in a plane;
- Dimensality is reduction to identifying trends in data set that operate along dimension that are not explicitly called out in the dataset
- Approaches for high dimensional dataset.
  - ↳ Missing value ratio
  - ↳ Low Variance filter
  - ↳ High correlation filter
  - ↳ Random forest
  - ↳ PCA
  - ↳ Backward feature
  - ↳ forward

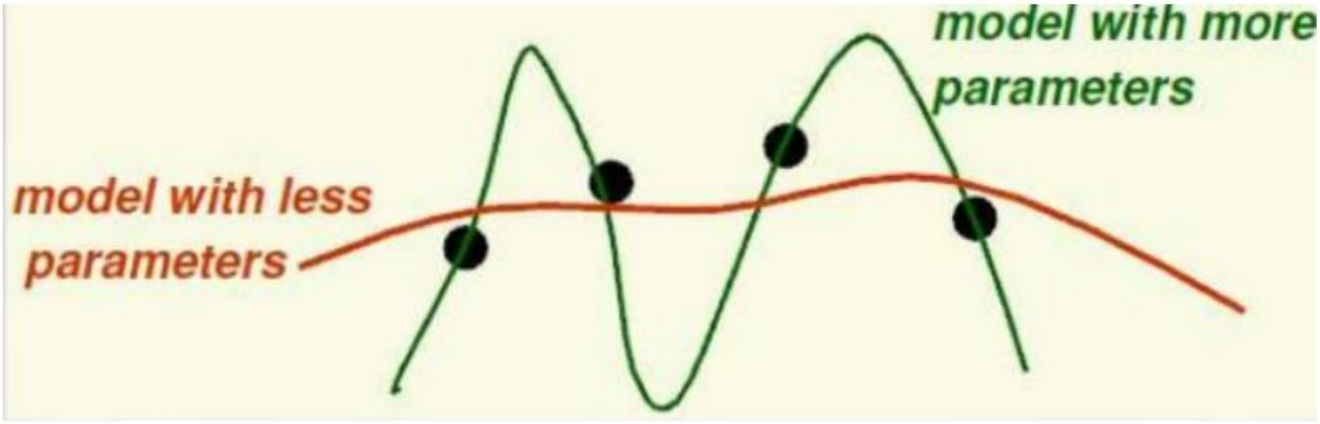
4) <u>What : Classification Or Regression ??</u>



Classification or Regression

- In underfitting, model performs poorly on training data.
- In overfitting, model performs well on training data but doesn't well on validation or testing data.
- Amount of regularization should be decreased & new specific features should be added in poor performance and consider fewer features combination to avoid overfitting.
- For higher accuracy, amount of data should be increase
- What basically the disadvantage of overfit & underfit?
- If a dataset is overfit, may be the outliers present in graph so the model will neglect due to overfitting.
- And if it is underfit it will neglect most of data set.
- So both the condition are useless.

5) Model : Linear vs. Non-Linear



Model: Linear Vs Nonlinear

The given model is linear b/c each term is either a constant or the product of the parameter & a product variable. In given data, nonlinear is curving many points forms, thats why non-linear regression provide flexible curve.

- The following picture demonstration points are discuss below:
- The Green line shows; basically it has more parameters and more data that's the ~~resist~~ reason for its precision & accuracy and given us exact ~~wire~~ varying curve ~~rather~~, overfitting line, rather.
- Red line having less parameter and less data record fo it, that is precisely the reason, for its straight ~~overfitting~~ underfitting line.

Week: 11

"ANOVA Table"

- Confusion matrix &
- Anova table

Date:

ASSIGNMENT

PRACTICE CONFUSION MATRIX

A confusion matrix is a technique for summarizing the performance of classification algorithm.

Classification accuracy alone can be misleading if you have an unequal no of observation in each class or if you have more than two classes in your dataset.

There are two things in confusion matrix.

1) Actual values
2) Predicted values.

eg. A patient have disease or not.

165	No	Yes	
No	50 TN	10 FP	60
Yes	5 FN	100 TP	105
	55	110	

- There are two posible predicted classes yes or no.
- The classifier made 165 predictions.
- Out of those 165 cases the predicted "yes" is 110 times & "No" 55 times & in reality 105 patients in the sample have diseases & 60 patients donot have disease.

SOLUTION:

We will find accuracy.

$$\text{Accuracy} = \frac{\text{Total correct prediction}}{\text{Total actual}}$$

$$= \frac{TP + TN}{\text{Total}}$$

$$= \frac{50 + 100}{165}$$

Accuracy = 0.90

Error = $\frac{\text{Total incorrectly predicted}}{\text{Total predictions}}$

$= \frac{17}{165}$

Error = 0.103

Precision = $\frac{\text{correctly predicted}}{\text{Total predited}}$

$= \frac{100}{110}$

Precision = 0.65

. Consider the following 3-class confusion matrix calc precision & recall per class & also calculate weig average precision & recall for classifier.

100	A	B	C	
A	15	2	3	20
B	7	15	8	30
C	2	3	45	50
	24	20	56	

SOLUTION:

Precision = $\frac{\text{correctly predited}}{\text{Total predicted}}$

Class A precision = $\frac{15}{24} = 0.625$

class B precision = $\frac{15}{20} = 0.75$

class C precision = $\frac{45}{56} = 0.80$

Recall = $\frac{\text{correctly classified}}{\text{Actual}}$

class A recall = $\frac{15}{30} = 0.75$

Date:

class B recall = $\frac{15}{30} = 0.5$

class C recall = $\frac{45}{50} = 0.9$

Accuracy = $\frac{\text{Total correctly classified}}{\text{Total actual}}$

$= \frac{15+15+45}{100}$

Accuracy of classifier = 0.75

- Why do we need confusion matrix?

1) It shows us how any classification model is confused when it makes prediction.
2) Confusion matrix not only gives you insight into the error being made by you classifier but also types of error that are being made.
3) Every column of the confusion matrix represents instances of the predicted class.
4) Every row of the confusion matrix represents instances of the actual class.

ANOVA TOY EXAMPLE:-

1) Three different techniques medication exercises & special diet are randomly assigned to (individuals diagnosed with high blood pressure) lower the blood pressure. After the four weeks the reduction in each person's blood pressure is recorded. Test at 5% level whether there is significant difference in mean reduction of blood pressure among the three techniques.

Medication	10	12	9	15	13
Exercise	6	8	3	0	2
Diet	5	9	12	8	4

Solution:

• Step 1: Hypothesis:

Noull hypothesis : $H_0 = \mu_1 = \mu_2 = \mu_3$

That is there is no significant difference among the group on the average reduction in blood pressure

Alternative hypothesis = $H_1 = \mu_i \neq \mu_j$ for atleast on pair (i,j); $ij = 1,2,3$; $i \neq j$

That is there is significant difference in the average reduction in blood pressure in atleast one pair of treatment

• Step 2: Data

Medication	10	12	9	15	13
Exercise	6	8	3	0	2
Diet	5	9	12	8	4

• Step 3: level of significance

$\alpha = 0.05$

• Step 4: Test statistics

$F_0 = MST/MSE$

• Step 5: Calculation of test statistics:

Total medication	59	3481
Total execuise	19	361
Total diet	39	1521
	$G = 116$	5286

• Individual square

Medication	100	144	81	225	169
Exercise	36	64	9	0	4
Diet	25	81	144	64	16

$\sum\sum x^2 = 1162$

1) Correction factor:

$$CF = \frac{G^2}{n} = \frac{(116)^2}{15} = \frac{13456}{15} = 897.06$$

2) Total sum of squares:

$TSS = \sum\sum x^2 - CF$

$= 1162 - 897.06$

$= 264.94$

3) Sum of squares b/w treatments:

$SST = \frac{\sum x^2}{n} - C.F$

$= \frac{5286}{5} - 897.06$

$= 1057.2 - 897.06$

$= 160.14$

4) Sum of squares due to error

$SSE = TSS - SST$

$= 264.94 - 160.14$

$= 104.8$

ANOVA TABLE:

Source of variation	Sum of squares	Degree of freedom	MST	F. rates
Treatment	SST 160.14	K-1=2	80.7	9.17
Error	SSE 104.8	n-K=12	8.73	
Total	TSS 264.8	n-1=14		

- Step 6: Critical value

$f(2,12), \alpha\ 0.05 = 3.8853$

- Step 7: Decision

As $F_0 = 9.17 > f(2,12)\ 0.05 = 3.8853$, the null hypothesis is rejected. Hence, we conclude that there exists significant difference in the reduction of the average blood pressure in atleast one pair of techniques.

2) Three composition instructor recorded the no of spelling errors which their students made on a research paper. At 1% level of significance test whether there is significant difference in the average number of errors in the three classes of the instructor.

Instructor 1	2	3	5	0	8			
Instructor 2	4	6	8	4	9	0	2	
Instructor 3	5	2	3	2	3	3		

Solution:

- Step 1: Hypothesis.

Null hypothesis ~~H~~ Ho: $\mu_1 = \mu_2 = \mu_3$?

That is there is no significant difference among the no of errors in the three columns of students.

Alternative hypothesis:

H1: $\mu_i \neq \mu_j$ for at one pair $(i,j) = 1,2,3$ $i \neq j$ That is one pair of groups differ significantly on the mean error.

- Step 2: Data.

Instructor 1	2	3	5	0	8			
Instructor 2	4	6	8	4	9	6	2	
Instructor 3	5	2	3	2	3	3		

- Step 3: level of significance

$\alpha = 5\%$

- Step 4: Test statistics.

$F_0 = MST / MSE$

- Step 5: Calculation of test statistics

								Total	$\bar{y}_i$
Instructor 1	2	3	5	0	8			18	324
Instructor 2	4	6	8	4	9	0	2	33	1089
Instructor 3	5	2	3	2	3	3		18	324
								69	

Date:

Individual square:

Instructor 1	9	9	25	0	64		
Instructor 2	16	36	64	16	81	0	4
Instructor 3	25	4	9	4	9	9	

$\Sigma\Sigma x^2 = 379$

1) Correction factor

$$CF = \frac{G^2}{n} = \frac{(69)^2}{18} = \frac{4761}{18} = 264.5$$

2) Total sum of square:

$$TSS = \Sigma\Sigma x^2 - C.F$$
$$= 379 - 264.5 = 114.5$$

3) Sum of squares b/w treatment:

$$SST = \frac{\Sigma x^2}{n} - C.F$$
$$= \left(\frac{324}{5} + \frac{1089}{7} + \frac{324}{5}\right) - 264.5$$
$$= (274.4) - 264.5$$
$$= 9.9$$

4) Sum of squares done to error:

$$SSE = TSS - SST$$
$$= 114.5 - 9.9$$
$$= 104.96$$

ANOVA TABLE:

Source of variation	Sum of square	Degree of freedom	MST	F-ratio
B/w treatment	9.9	2	4.95	$\frac{4.95}{6.97} = 0.710$
Error	104.6	15	6.97	
Total	114.5	17		

- Step 6: Critical value:

The critical value $f(15, 2)\ 0.05 = 3.6823$

• Step 7 : decision.

As $F_0 = 0.710 < f(1,52)_{0.05} = 3.6823$ the null hypothesis is not rejected. There is not enough evidence to reject the null hypothesis & hence we conclude the mean number of error made by these classes of student are not

CONFUSION MATRIX TOY EXAMPLE:

1) Consider the example & calculate accuracy, recall

F1:

	Predicted Yes	Predicted No	
280	TP 150	FN 10	160
Actual Yes	FP 20	TN 100	120
Actual no	170	110	

Solution:

$$\text{Accuracy} = \frac{TP + TN}{\text{Total}}$$

$$= \frac{150+100}{280}$$

$$\text{Accuracy} = 089$$

$$\text{Recall} = \frac{TP}{\text{Actual Yes}} = \frac{150}{160}$$

$$\text{Recall} = 0.93$$

$$\text{Precision} = \frac{TP}{\text{Predited Yes}} = \frac{150}{170}$$

$$\text{Precision} = 0.88$$

$$F1 = \frac{2 \times \text{recall} \times \text{Precision}}{\text{recall} + \text{precision}}$$

$$= \frac{2 \times 0.93 \times 0.88}{0.93 + 0.88}$$

$$F1 = 0.90$$

K-MEAN TOY EXAMPLE:-

We have several of objects (4 types of medicine) & each object have two ~~dis~~ attributes or features as shown in the table below:

Our goal is to group these objects into K=2 group of medicine based on the two features (PH & weight index)

Object	(attribute 1 (x) index	attribute 2 (Y))
A	1	1
B	2	1
C	4	3
D	5	4

Solutions:-

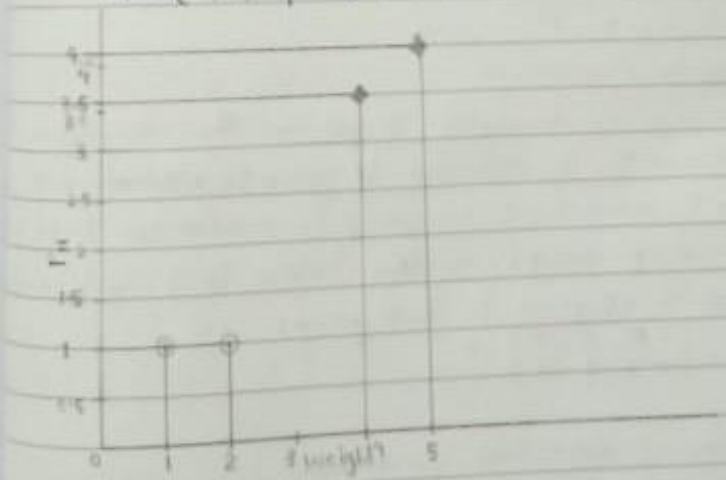
1) Initial value of centroid:

So suppose we have used med A & B as first centroid.

Let C_1 & C_2 donate the coordinates of the centroid

So,

$C_1 = (1,1)$ & $C_2 = (2,1)$



2) Object centroid distance:

We calculate the distance b/w cluster centroid to each object.

Lets' use Euclidean distance, then we have distance

matrix at iteration 1:

$$D^0 = \begin{bmatrix} 0 & 1 & 3.61 & 5 \\ 1 & 0 & 2.83 & 4.24 \end{bmatrix}$$

$C_1 = (1,1)$

$C_2 = (2,1)$

$$\begin{bmatrix} A & B & C & D \\ 1 & 2 & 4 & 5 \\ 1 & 1 & 3 & 4 \end{bmatrix} \begin{matrix} x \\ y \end{matrix}$$

Each column in the distance matrix symbolize the object. The first row of the distance matrix corresponds to the distance of each object to the first centroid & the second row is the distance of each object to the second centroid.

$C_1 = (1,1)$

E.D of first centroid:

$D^0 = \sqrt{(4-1)^2 + (3-1)^2} = 3.61$

$C_2 = (2,1)$

E.D of second centroid:

$D^0 = \sqrt{(4-2)^2 + (3-1)^2} = 2.83$

3) Objects clustering:

We assign each object based on the minimum d... The medicine A is assigned to group 1, medicine B ... to group 2, medicine C to group 2, medicine D to ... The element of group1 matrix below is 1 if & only ... the object is assigned to that group:

$$G^0 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 \end{bmatrix} \begin{matrix} g1 \\ g2 \end{matrix}$$

4) Iteration 1 determines centroids:

Group 1 only has one member thus, the centroid ... remain in $C_1(1,1)$. Group2 has now three mem... thus centroid is the average coordinate:

$$C_2 = \left(\frac{2+4+5}{3}, \frac{1+3+4}{3} \right)$$

$C_2 = \left(\frac{11}{3}, \frac{8}{3}\right)$

Iteration 1 object centroids distances:
The next step is to compute the distance of all objects to new centroids:

$$D^0 = \begin{bmatrix} 0 & 1 & 3.61 & 5 \\ 3.14 & 2.36 & 0.47 & 1.89 \end{bmatrix} \begin{matrix} C_1 = (1,1) \\ C_2 = (11/3, 8/3) \end{matrix}$$

$$\begin{matrix} A & B & C & D \end{matrix}$$
$$\begin{bmatrix} 1 & 2 & 4 & 5 \\ 1 & 1 & 3 & 4 \end{bmatrix} \begin{matrix} x \\ y \end{matrix}$$

Iteration 1. objects clustering:
we assign each object based on the minimum distance Based on the new distance matrix, we now move the med B to group 1 while all the other objects remain same. The group matrix is shown below;

$$G' = \begin{bmatrix} 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \end{bmatrix} \begin{matrix} G-1 \\ G-2 \end{matrix}$$

Iteration 2 determines centroids:
Calculate the new centroid coordinates based on the clustering of previous iteration group 1 & group 2 both have two members, so

$$C_1 = \left(\frac{1+2}{2}, \frac{1+1}{2}\right), \quad C_2 = \left(\frac{4+5}{2}, \frac{3+4}{2}\right)$$

$$C_1 = \left(1\frac{1}{2}, 1\right), \quad C_2 = \left(4\frac{1}{2}, 3\frac{1}{2}\right)$$

Iteration 2 object-centroid:
we have new distance matrix at iteration 2 as,

$$D^2 = \begin{bmatrix} 0.5 & 0.5 & 3.20 & 4.61 \\ 4.30 & 3.54 & 0.71 & 0.71 \end{bmatrix} \begin{matrix} C_1 = \left(1\frac{1}{2}, 1\right) \\ C_2 = \left(4\frac{1}{2}, 3\frac{1}{2}\right) \end{matrix}$$
$$\begin{bmatrix} 1 & 2 & 4 & 5 \\ 1 & 1 & 3 & 4 \end{bmatrix} \begin{matrix} X \\ Y \end{matrix}$$

Date:

9) Iteration-2 object clustering

We assign each object based on the maximum distance:

$$G^2 = \begin{bmatrix} 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \end{bmatrix} \begin{matrix} G-1 \\ G-2 \end{matrix}$$

We obtain the result that $G^2 = G^1$.

comparing the grouping of last iteration & this iteration reveals that the object does not have group anymore.

The computation of the k-mean ~~of~~ clustering has reached its stability & no more iteration is needed.

We get the final grouping as the result:

Med A	1	1	1
Med B	2	1	1
Med C	4	3	2
Med D	5	4	2

Week: 12

"Feature Space concept K-means"

- Normalization

PRACTICE EXERCISE: NORMALIZATION

1) Show all the working steps for any three rows of given table:

input	0·0	0·1	0·2	0·3	0·4	0·5
Standardized	-1·33630	-0·80178	-0·26726	0·26126	0·80784	1·33634
normalized	0·0	0·2	0·4	0·6	0·8	1·0

Solution

Standardization

$$x = \frac{x - \text{mean}(x)}{\text{Standard deviation}}$$

Min-Max Normalization

$$x_{norm} = \frac{x - \min(x)}{\max(x) - \min(x)}$$

For Min-Max normalization:

For input 1·0:

$$x_{norm} = \frac{1.0 - 0.0}{5.0 - 0.0}$$

$$= 0.2$$

For input 2·0:

$$x_{norm} = \frac{2.0 - 0.0}{5.0 - 0.0}$$

$$= 0.4$$

For input 5·0:

$$x_{norm} = \frac{5.0 - 0.0}{5.0 - 0.0}$$

$$= 1.0$$

For Standardization:

$$x_{stan} = \frac{x - \text{mean}(x)}{\text{Standard deviation}}$$

For input 1·0

finding mean of x

$$\text{mean } (\bar{x}) = \frac{0.0+1.0+2.0+3.0+4.0+5.0}{6}$$

$$= 2.5$$

Finding S.D of x,

$$S.D = \sqrt{\frac{\Sigma(x-\bar{x})^2}{n-1}}$$

$$= \sqrt{\frac{(0.0-2.5)^2+(1.0-2.5)^2+(2.0-2.5)^2+(3.0-2.5)^2+(4.0-2.5)^2+(5.0-2.5)^2}{6-1}}$$

$$= 1.8708$$

Now,

$$X_{stan} = \frac{1.0-2.5}{1.8708}$$

$$= -0.8017$$

For input 2.0,

$$X_{stan} = \frac{2.0-2.5}{1.8708}$$

$$= -0.2672$$

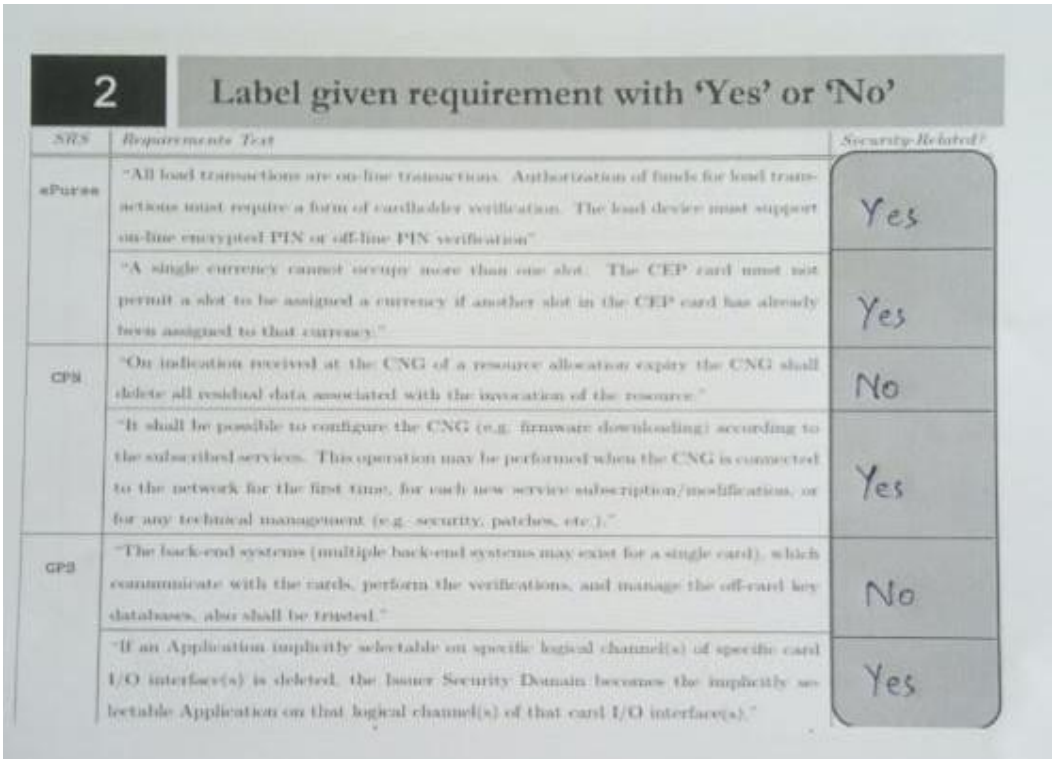
For input 5.0,

$$X_{stan} = \frac{5.0-2.5}{1.8708}$$

$$= 1.3363$$

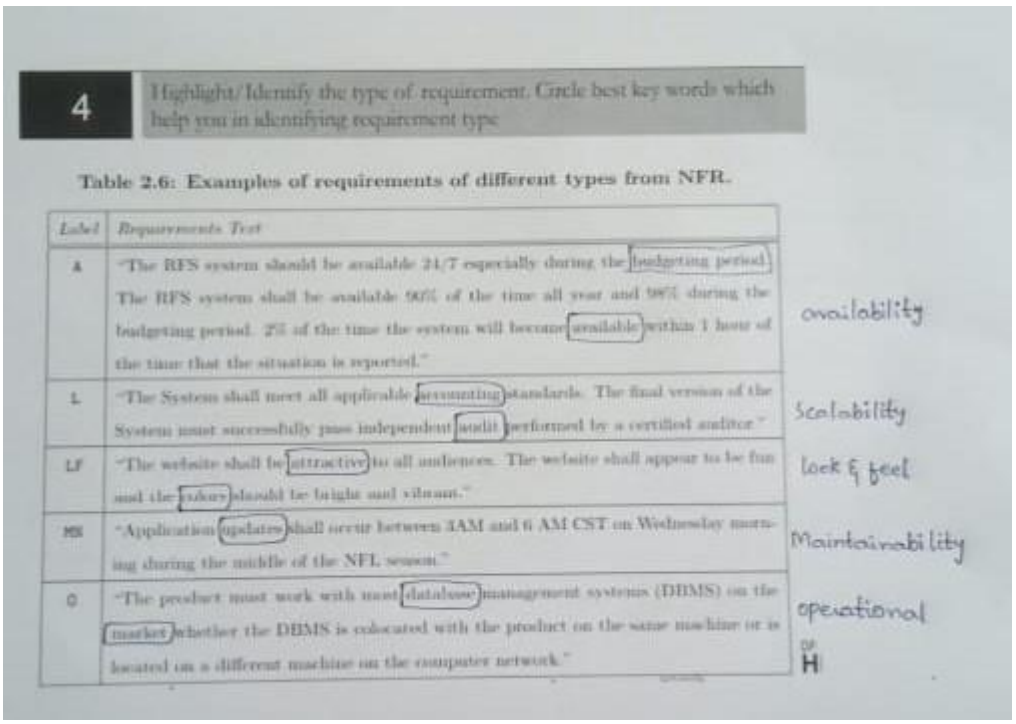
Week: 15

"Naive Bayes, Perceptron Rule Exam Discussion"



2 Label given requirement with 'Yes' or 'No'

SRS	Requirements Text	Security-Related?
ePurse	"All load transactions are on-line transactions. Authorization of funds for load transactions must require a form of cardholder verification. The load device must support on-line encrypted PIN or off-line PIN verification"	Yes
	"A single currency cannot occupy more than one slot. The CEP card must not permit a slot to be assigned a currency if another slot in the CEP card has already been assigned to that currency."	Yes
CPN	"On indication received at the CNG of a resource allocation expiry the CNG shall delete all residual data associated with the invocation of the resource."	No
	"It shall be possible to configure the CNG (e.g. firmware downloading) according to the subscribed services. This operation may be performed when the CNG is connected to the network for the first time, for each new service subscription/modification, or for any technical management (e.g. security, patches, etc.)."	Yes
GP2	"The back-end systems (multiple back-end systems may exist for a single card), which communicate with the cards, perform the verifications, and manage the off-card key databases, also shall be trusted."	No
	"If an Application implicitly selectable on specific logical channel(s) of specific card I/O interface(s) is deleted, the Issuer Security Domain becomes the implicitly selectable Application on that logical channel(s) of that card I/O interface(s)."	Yes



4 Highlight/Identify the type of requirement. Circle best key words which help you in identifying requirement type

Table 2.6: Examples of requirements of different types from NFR.

Label	Requirements Text	
A	"The RFS system should be available 24/7 especially during the budgeting period. The RFS system shall be available 90% of the time all year and 98% during the budgeting period. 2% of the time the system will become available within 1 hour of the time that the situation is reported."	availability
L	"The System shall meet all applicable accounting standards. The final version of the System must successfully pass independent audit performed by a certified auditor."	Scalability
LF	"The website shall be attractive to all audiences. The website shall appear to be fun and the colors should be bright and vibrant."	look & feel
MS	"Application updates shall occur between 3AM and 6 AM CST on Wednesday morning during the middle of the NFL season."	Maintainability
O	"The product must work with most database management systems (DBMS) on the market whether the DBMS is colocated with the product on the same machine or is located on a different machine on the computer network."	operational

DP H

5 Highlight/Identify the type of requirement. Circle best key words which help you in identifying requirement type

PE	"The search for the preferred repair facility shall take no longer than 8 seconds. The preferred repair facility is returned within 8 seconds."	Performance
SC	"The system shall be expected to manage the nursing program curriculum and class/ clinical scheduling for a minimum of 5 years."	Usability
SE	"The product shall ensure that it can only be accessed by authorized users. The product will be able to distinguish between authorized and unauthorized users in all access attempts."	Security
US	"If projected the data must be readable. On a 10x10 projection screen 90% of viewers must be able to read Event / Activity data from a viewing distance of 30."	Look & feel
F	"System shall automatically update the main page of the website every Friday and show the 4 latest movies that have been added to the website."	Legal

PRACTICE EXERCISE: FREQUENCY TABLE

Weather	Play
Sunny	No
Overcast	Yes
Rainy	Yes
Sunny	Yes
Sunny	Yes
overcast	Yes
rainy	No
rainy	No
sunny	Yes
rainy	Yes

Variable = weather

States { sunny, overcast, rainy, - - - }

Frequency table

weather	No	Yes
Overcast	-	4
Rainy	3	2
Sunny	2	3
Grand total	5	9

Model variables : 5 more : states

Outlook	Temp	Humidity	windy	Play Golf
Rainy	Hot	High	False	No
Rainy	Hot	High	True	No
overcast	Hot	High	False	Yes
sunny	Mild	High	False	Yes
sunny	Cool	Normal	False	Yes
sunny	Cool	Normal	True	No
overcast	Cool	Normal	True	Yes
rainy	Mild	High	False	NO
rainy	Cool	Normal	False	Yes
sunny	Mild	Normal	False	Yes
rainy	Mild	Normal	True	yes
overcast	Mild	High	True	Yes
overcast	Hot	Normal	False	Yes
sunny	Mild	High	True	No

Frequency table for Predictors:

Frequency table		Play golf	
		Yes	No
outlook	Sunny	3	2
	Overcast	4	0
	Rainy	2	3

Frequency table		Play golf	
		Yes	No
Temp	Hot	2	2
	Mild	4	2
	Cool	3	1

Frequency Table		Play golf	
		Yes	No
Humidity	High	3	4
	Normal	6	1

Frequency table		Play golf	
		Yes	No
windy	False	6	2
	True	3	3

PRACTICE EXERCISE: LIKELIHOOD TABLE

Weather	Play
Sunny	No
Overcast	Yes
Rainy	Yes
Sunny	Yes
Sunny	Yes
Overcast	Yes
Rainy	No
Rainy	No
Sunny	Yes
Rainy	Yes
Sunny	No
Overcast	Yes
Overcast	Yes
Rainy	No

Fxequency table

Weather	No	Yes
Overcast	-	4
Rainy	3	2
Sunny	2	3
Grand Total	5	9

likelihood table

weather	No	Yes		
Overcast	-	4	=4/14	0.29
Rainy	3	2	=5/14	0.36
Sunny	2	3	=5/14	0.36
All	5	9		
	=5/14	=9/14		
	0.36	0.64		

likelihood probability for outlook

$P(x|c) = P(\text{Sunny}|\text{Yes}) = 3/9 = 0.33$

Frequency table		Play golf: Yes	Play golf: NO
outlook	Sunny	3	2
	overcast	4	0
	Rainy	2	3

→

likelihood table		Play golf: Yes	Play golf: No	
outlook	Sunny	3/9	2/5	5/14
	overcast	4/9	0/5	4/14
	Rainy	2/9	3/5	5/14
		9/14	5/14	

$P(x) = P(\text{Sunny}) = 5/14 = 0.36$

$P(c) = P(\text{Yes}) = 9/14 = 0.64$

likelihood probability for other attributes

Frequency ~~likelihood~~ table		Play golf: Yes	Play golf: No
Humidity	High	3	4
	Normal	6	1

→

likelihood table		Play Golf: Yes	Play Golf: No
Humidity	high	3/9	4/5
	Normal	6/9	1/5

Date:

Frequency table		Play golf Yes	No
Temp	Hot	2	2
	Mild	4	2
	Cool	3	1

likelihood table		Play golf Yes	No
Temp	Hot	2/9	2/5
	Mild	4/9	2/5
	Cool	3/9	1/5

Frequency table		Play golf Yes	No
windy	False	6	2
	True	3	3

→

likelihood table		Play golf Yes	No
windy	False	6/9	2/5
	True	3/9	3/5

PRACTICE EXERCISE: POSTERIOR PROBABILITY

Posterior probability of playing golf (Yes)

Frequency table		Play golf Yes	No
outlook	Sunny	3	2
	Overcast	4	0
	Rainy	2	3

→

likelihood table		Play golf Yes	No	
outlook	Sunny	3/9	2/5	5/14 → P(x) = P(Sunny) = 5/14 = 0.36
	overcast	4/9	0/5	4/14
	Rainy	2/9	3/5	5/14
		9/14	5/14	

P(x|C) = P(Sunny|Yes) = 3/9 = 0.33

P(C) = P(Yes) = 9/14 = 0.64

Posterior probability = $P(C|x) = P(Yes|Sunny)$

$$= \frac{0.33 \times 0.64}{0.36} = 0.60$$

Posterior probability of Not playing golf (No)

Frequency Table		Play golf Yes	No
outlook	sunny	3	2
	overcast	4	0
	rainy	2	3

→

likelihood table		Play golf Yes	No	
outlook	sunny	3/9	2/5	5/14 → P(x) = P(sunny) = 5/14 = 0.36
	overcast	4/9	0/5	4/14
	rainy	2/9	3/5	5/14
		9/14	5/14	

P(x|C) = P(Sunny|No) = 2/5 = 0.4

P(C) = P(No) = 5/14 = 0.36

Posterior probability = P(C|x)

$= P(No|Sunny) = \frac{0.40 \times 0.36}{0.36} = 0.40$

The

The class with the highest posterior probability is the outcome of prediction.

PRACTICE EXERCISE: DOT PRODUCT & WEIGHT

PERCEPTRON LEARNING APPROACH

A general algorithm for supervised learning follows:

Make an initial guess for each component of w,
Select a training set of data.
For each vector in the training set:
Compute $D(x)$
If $D(x) > 0$ & $x \in$ class 1 or $D(x) < 0$ & $x \in$ class 2, do not adjust w

- If $D(x) > 0$ & $x \in$ class 2 adjust w according to rule 1
- If $D(x) < 0$ & $x \in$ class 1 adjust w according to rule 2

until w does not change (or until criterion function is minimized).

2 class problem: linear or Non-linear

Table: Feature vector values for differentiation b/w Myocardial Infarction (MI) & Angina

Feature vector	Diagnosis	Systolic BP	White blood count
x_1	MI	110	13000
x_2	MI	90	12000
x_3	MI	85	18000
x_4	MI	120	8000
x_5	MI	130	18000
x_6	Angina	180	5000
x_7	Angina	200	7500
x_8	Angina	165	6000
x_9	Angina	190	6500
x_{10}	Angina	120	9000

Initial guess weight vector & dot product ($D(x) = w \cdot x = w_1x_1 + w_2x_2$)

$t_1 = (11.0, 13.0)$ (vector x_1, class 1)

$t_2 = (18.0, 5.0)$ (vector x_6, class 2)

$t_3 = (9.0, 12.0)$ (vector x_2, class 1)

$t_4 = (20.0, 7.5)$ (vector x_7, class 2)

We will make an initial guess for each weight as $w_1 = 0.3$, $w_2 = 1.0$. Initially, we substitute vector t_1 into eq (1.5):

$D(t_1) = -0.3(11.0) + 1.0(13) > 0$ therefore $y(t) = 1$

t_1 belongs to class 1; therefore $d(t) = 1$

Continue till convergence:

$D(t_4) = -0.3(20.0) + 1.0(7.5) > 0, y(t) = 1$

t_4 belongs to class 2

Therefore, substituting into eq (1.6):

$w_1(1) = -0.3 + 0.01[(-1 - (1)]20 = 0.7$

$w_2(1) = 1.0 + 0.01[-1 - (1)]7.5 = 0.85$

The process must then begin with t_1 & continue until all vectors are classified correctly. After completion of this process, the resulting weights are;

$w_1 = -0.7$

$w_2 = 0.85$

Our decision surface is

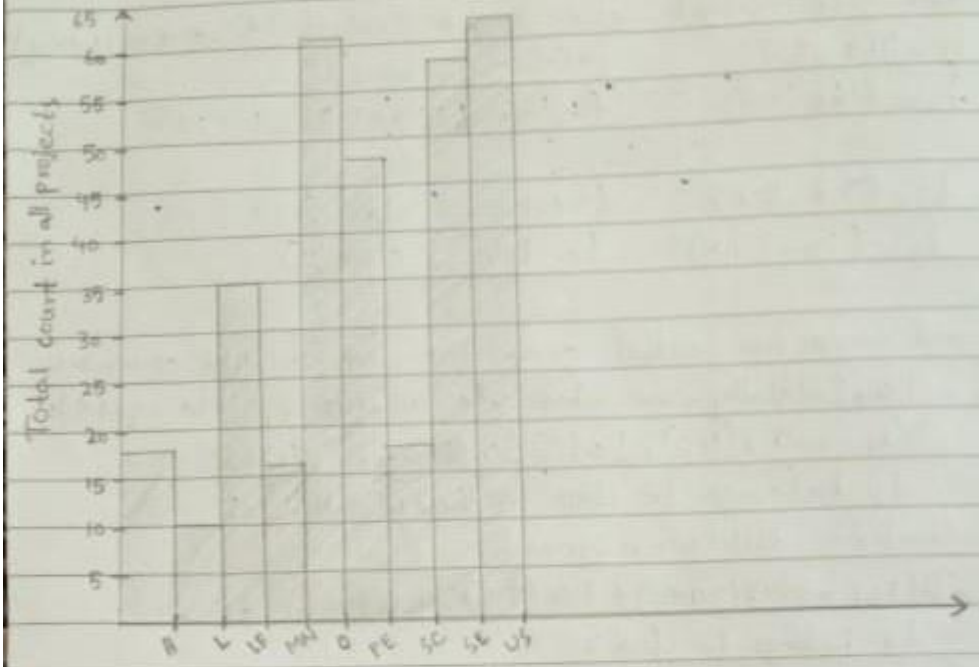
$$D(x) = -0.7x_1 + 0.85x_2 \quad (1.7)$$

PRACTICE EXERCISE: HISTOGRAM & CLUSTER:

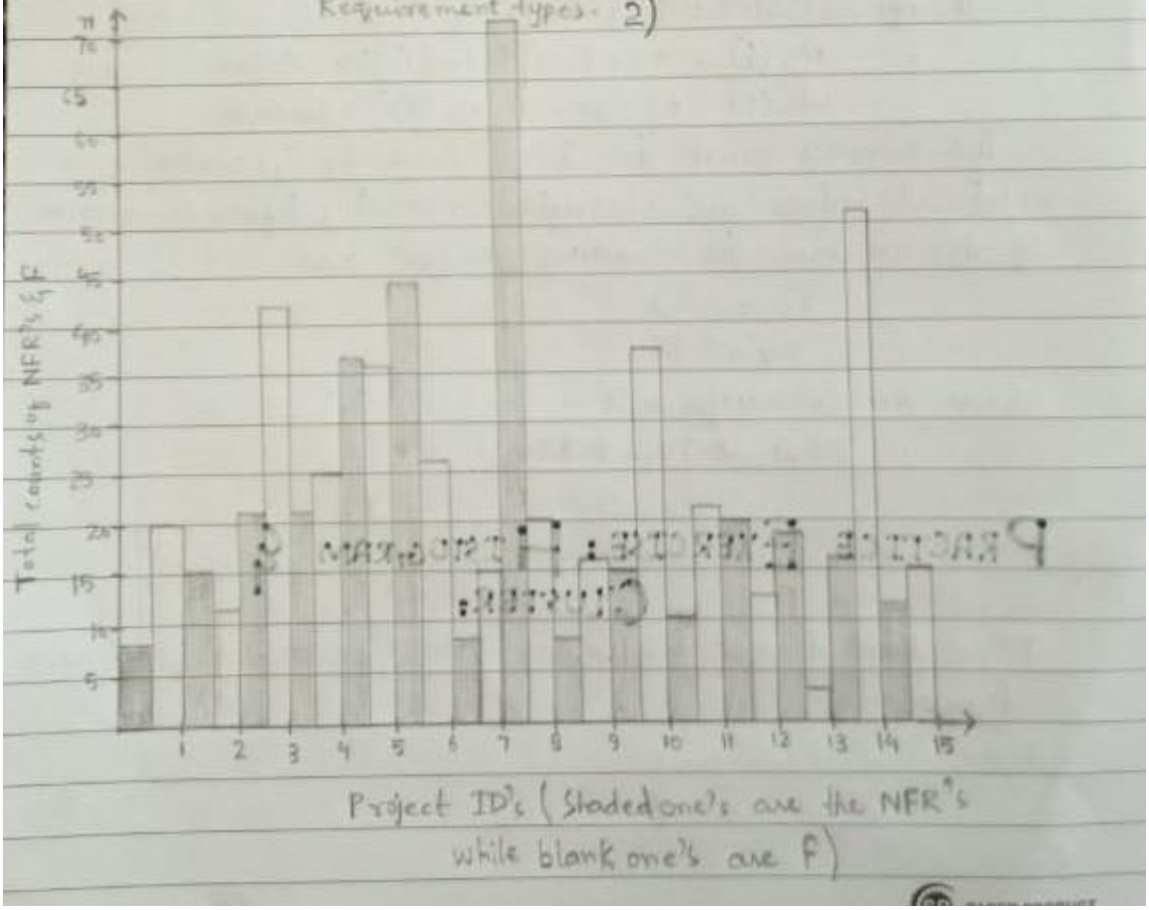
Table NFR dataset, broken down by project & requirement types:

Histograms

1)



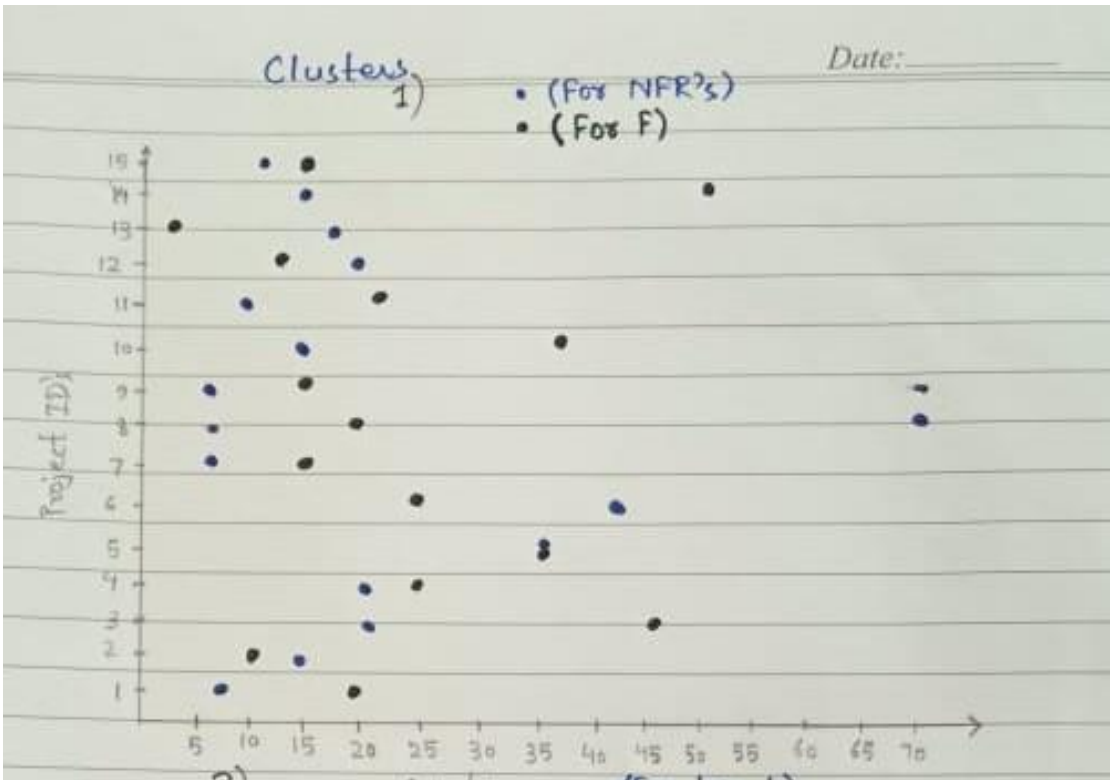
2)



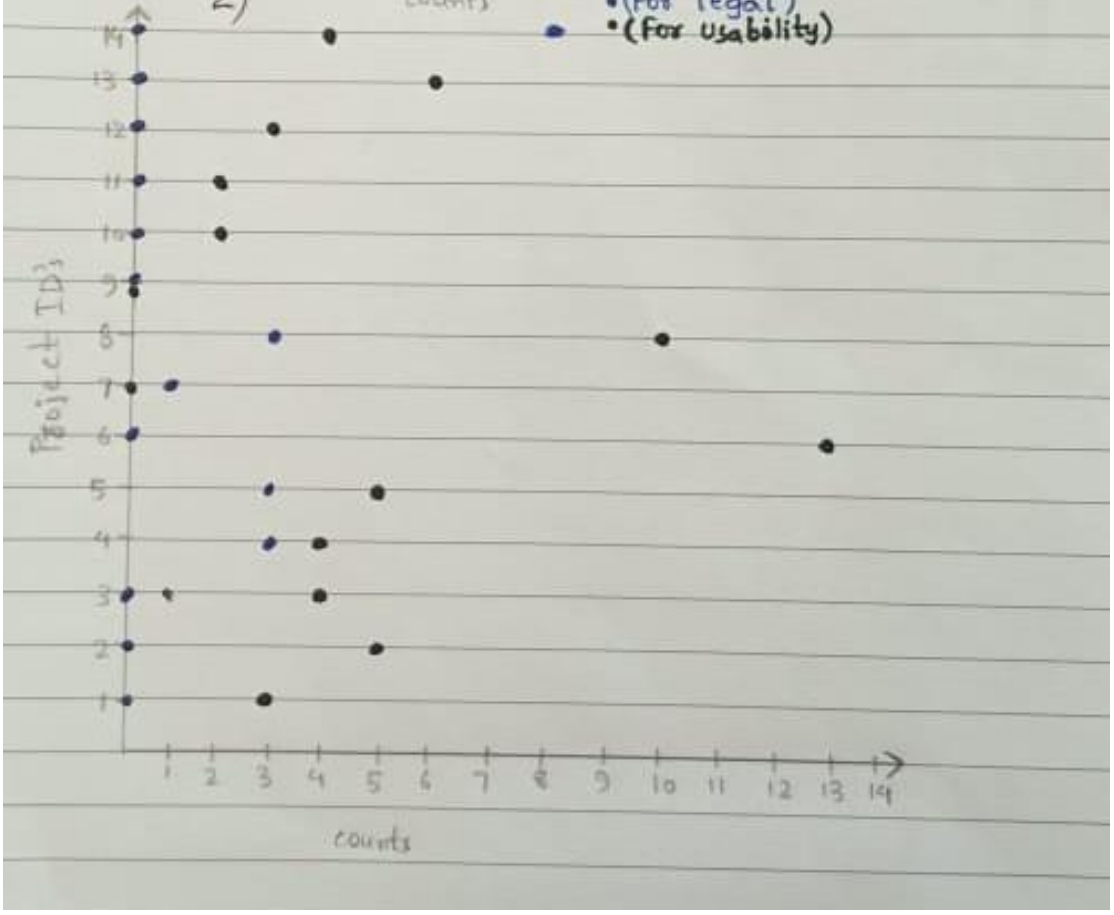
Date:

Clusters

1)



2)



Difference between Panda and NumPy libraries

Panda	NumPy
When we work on **Tabular data**, we prefer pandas module.	When we work on **Numerical data**, we prefer the NumPy module.
The powerful tools of pandas are **Data frame and Series.**	The powerful tool of NumPy is **Arrays.**
Pandas offers 2d table object called **Data Frame.**	NumPy is capable of providing **multi-D arrays.**
Pandas **consume more memory**.	NumPy is **memory efficient.**

Data train for Dog breed identifier

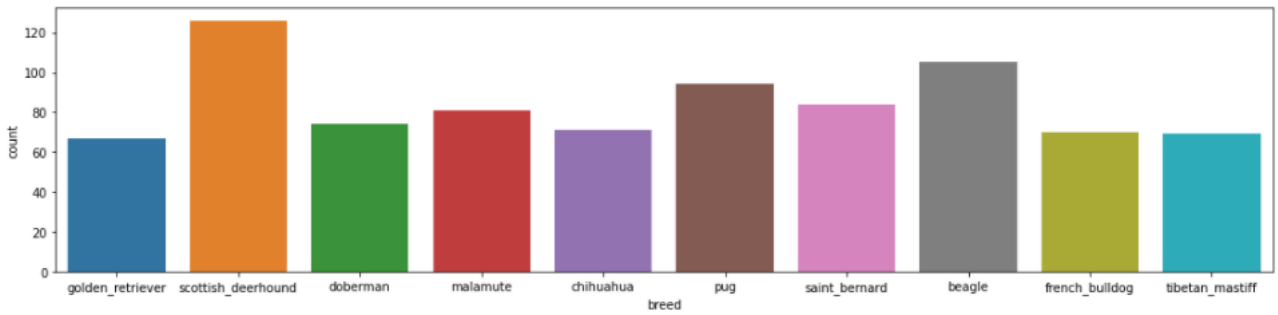
Dataset from Kaggle: https://www.kaggle.com/c/dog-breed-identification/code

In [2]:
```
#importing libraries
import numpy as np
import tensorflow as tf
from keras.applications.resnet_v2 import ResNet50V2
from tensorflow import keras
from keras.preprocessing.image import ImageDataGenerator
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
```

In [3]:
```
#defining directories
train_path = '../input/dog-breed-identification/train'
test_path = '../input/dog-breed-identification/test'

#reading dataset labels
train_labels = pd.read_csv('../input/dog-breed-identification/labels.csv')
test_labels = pd.read_csv('../input/dog-breed-identification/sample_submission.csv')
```

In [4]:
```
train_labels.head()
```

In [8]:
```
plt.figure(figsize=(18,4))
cp = sns.countplot(x = 'breed', data = train_labels)
```



Note: this model accuracy and lose feature gives error