selected neurons are ignored during training.

Handling Underfitting:

- Get more training data.
- Reduce the capacity of the network.
- Add weight regularization.
- Add dropout.

With these techniques, you should be able to improve your models and correct any overfitting or underfitting issues.

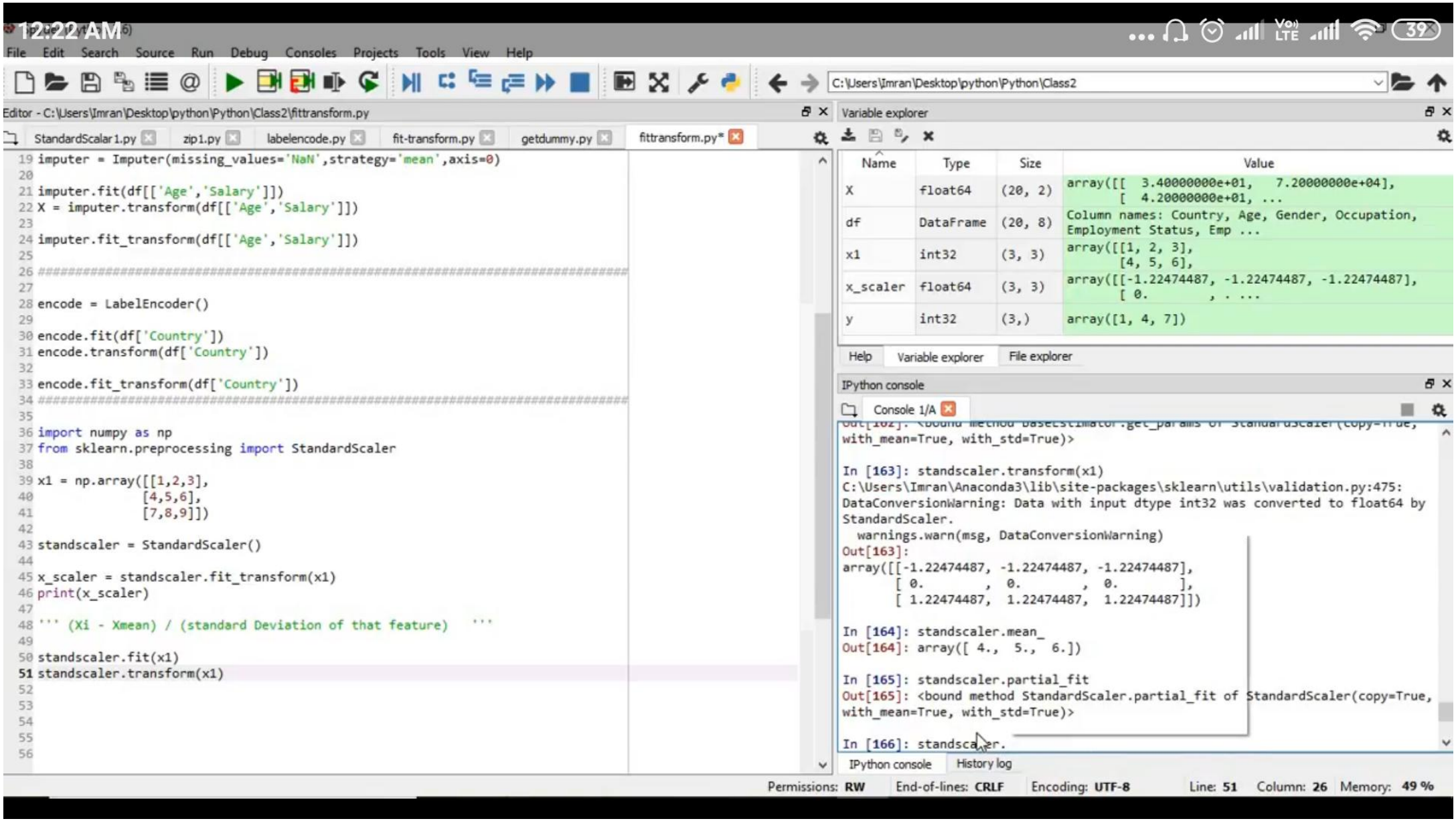
Connect with me on:

Håkon Hapnes Strand, Data Scientist

Answered Mar 11, 2018

The idea behind cross-validation is the same as with a single holdout validation set, to estimate the model's predictive performance on unseen data. Cross-validation simply does this more robustly, by repeating the experiment multiple times, using all the different parts of the training set as validation sets. This gives a more accurate indication of how well the model generalizes to unseen data.

In other words, cross-validation does not prevent overfitting in itself, but it may help in identifying a case of overfitting.

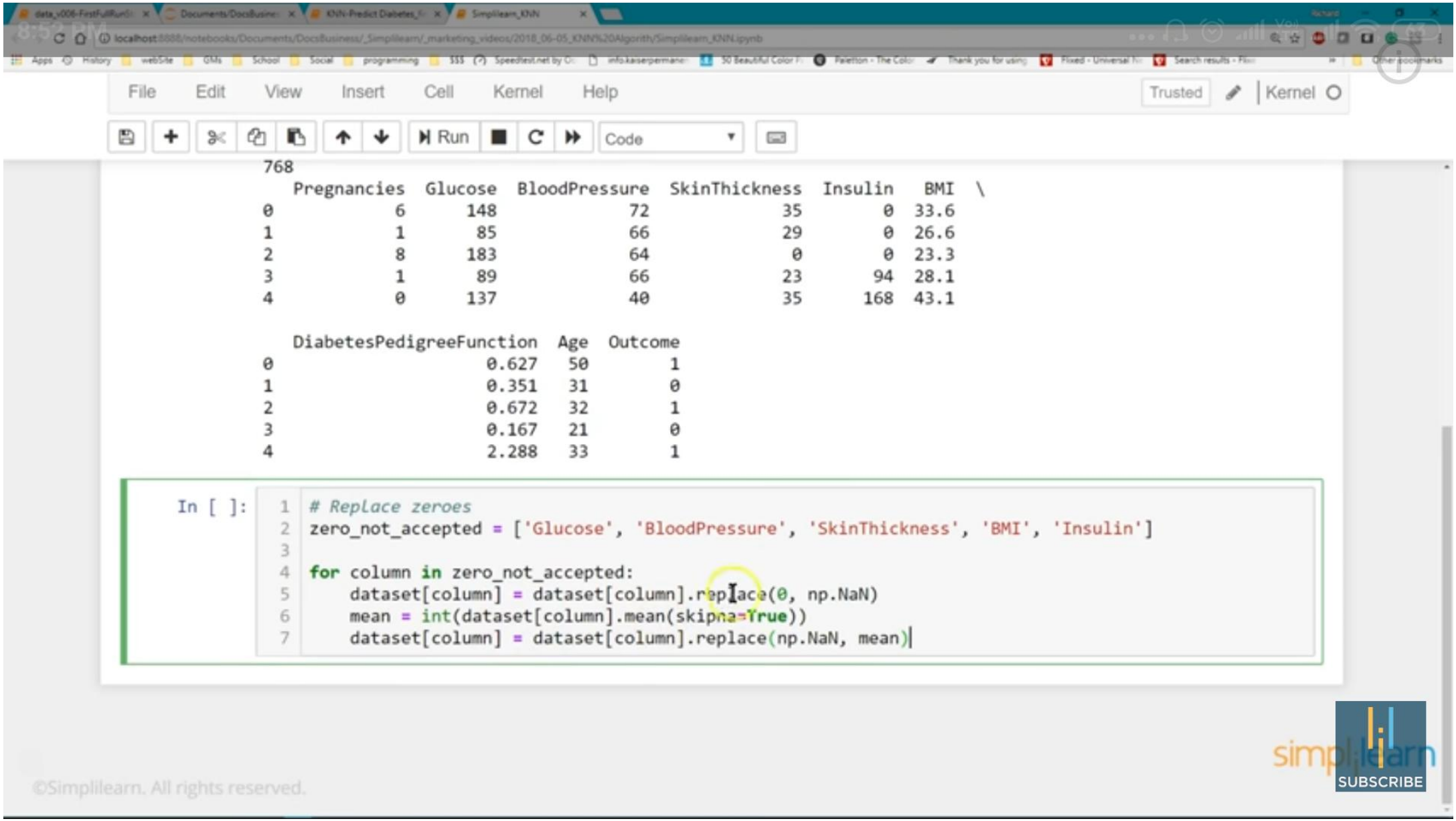

12:22 AM
File Edit Search Source Run Debug Consoles Projects Tools View Help
C:\Users\Imran\Desktop\python\Python\Class2
Editor - C:\Users\Imran\Desktop\python\Python\Class2\fittransform.py
StandardScalar 1.py
zip1.py
labelencode.py
fit-transform.py
getdummy.py
fittransform.py*
imputer = Imputer(missing_values='NaN',strategy='mean',axis=0)
imputer.fit(df[['Age','Salary']])
X = imputer.transform(df[['Age','Salary']])
imputer.fit_transform(df[['Age','Salary']])
encode = LabelEncoder()
encode.fit(df['Country'])
encode.transform(df['Country'])
encode.fit_transform(df['Country'])
import numpy as np
from sklearn.preprocessing import StandardScaler
x1 = np.array([[1,2,3],
[4,5,6],
[7,8,9]])
standscaler = StandardScaler()
x_scaler = standscaler.fit_transform(x1)
print(x_scaler)
''' (Xi - Xmean) / (standard Deviation of that feature) '''
standscaler.fit(x1)
standscaler.transform(x1)
Variable explorer
Name Type Size Value
X float64 (20, 2) array([[3.40000000e+01, 7.20000000e+04],
[4.20000000e+01, ...
df DataFrame (20, 8) Column names: Country, Age, Gender, Occupation, Employment Status, Emp ...
x1 int32 (3, 3) array([[1, 2, 3],
[4, 5, 6],
x_scaler float64 (3, 3) array([[-1.22474487, -1.22474487, -1.22474487],
[0. , ...
y int32 (3,) array([1, 4, 7])
Help
Variable explorer
File explorer
IPython console
Console 1/A
with_mean=True, with_std=True)>
In [163]: standscaler.transform(x1)
C:\Users\Imran\Anaconda3\lib\site-packages\sklearn\utils\validation.py:475: DataConversionWarning: Data with input dtype int32 was converted to float64 by StandardScaler.
warnings.warn(msg, DataConversionWarning)
Out[163]:
array([[-1.22474487, -1.22474487, -1.22474487],
[0. , 0. , 0.],
[1.22474487, 1.22474487, 1.22474487]])
In [164]: standscaler.mean_
Out[164]: array([4., 5., 6.])
In [165]: standscaler.partial_fit
Out[165]: <bound method StandardScaler.partial_fit of StandardScaler(copy=True, with_mean=True, with_std=True)>
In [166]: standscaler.
IPython console
History log
Permissions: RW
End-of-lines: CRLF
Encoding: UTF-8
Line: 51
Column: 26
Memory: 49 %

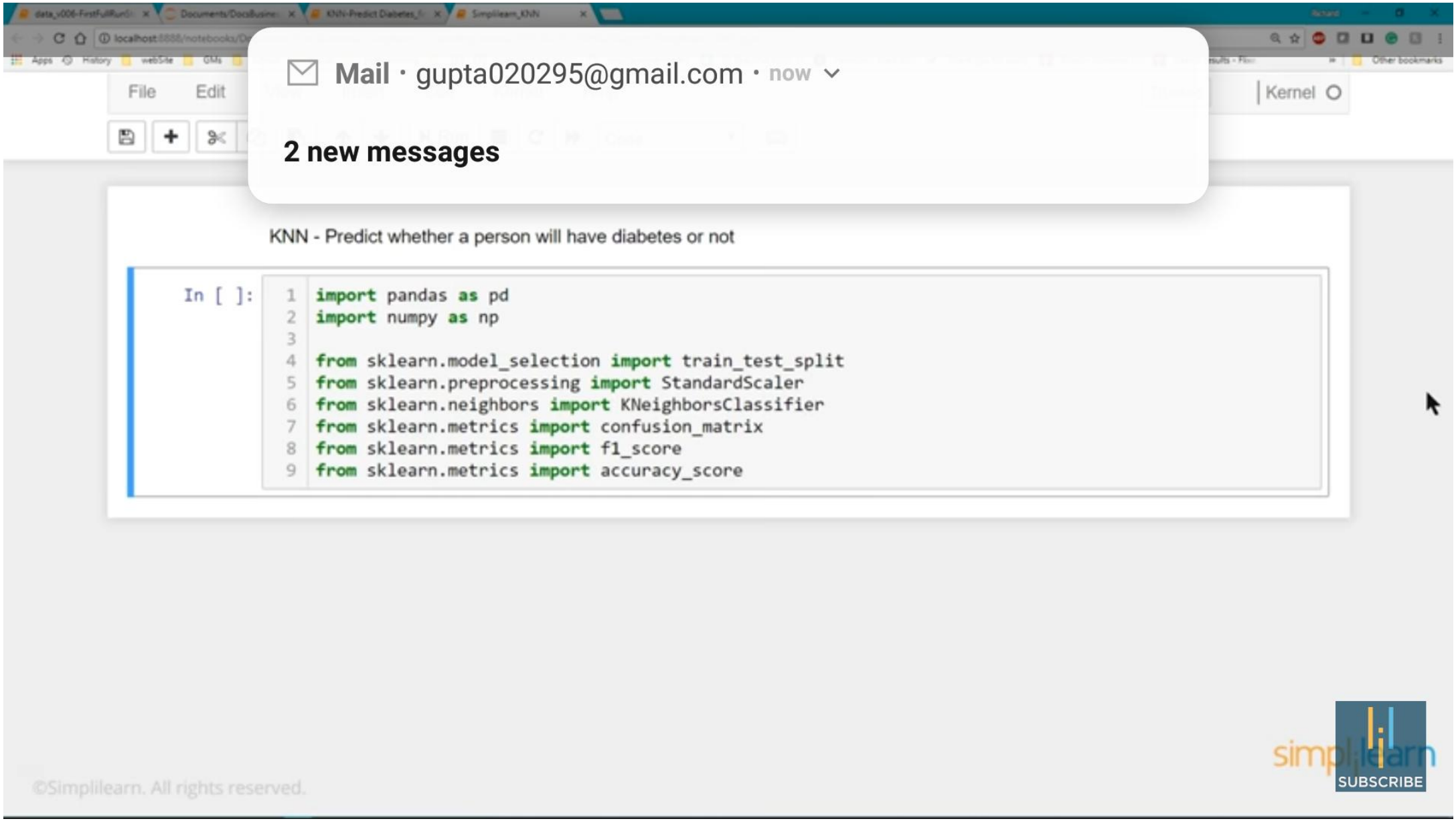
KNN - Predict diabetes

Rule of thumb: Any algorithm that computes distance or assumes normality, **scale your features!**

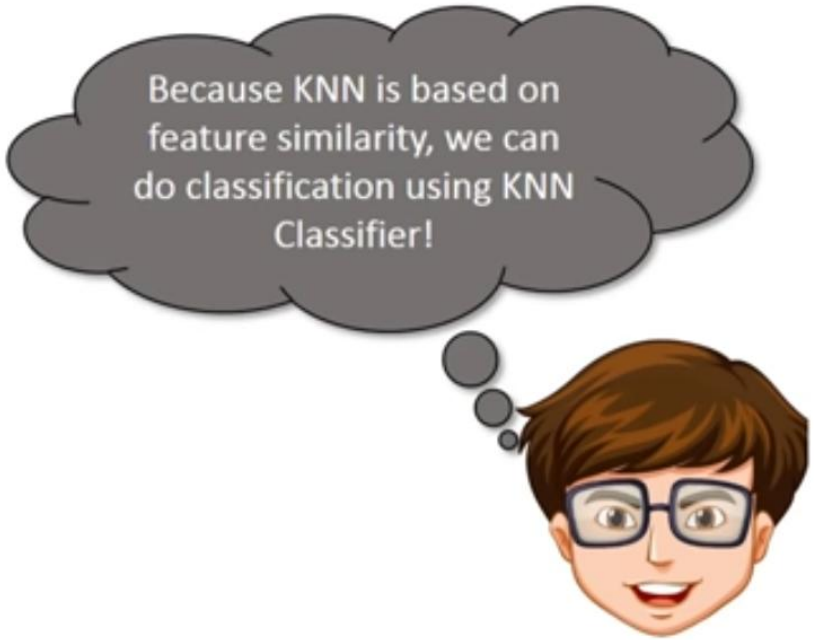
Feature Scaling:

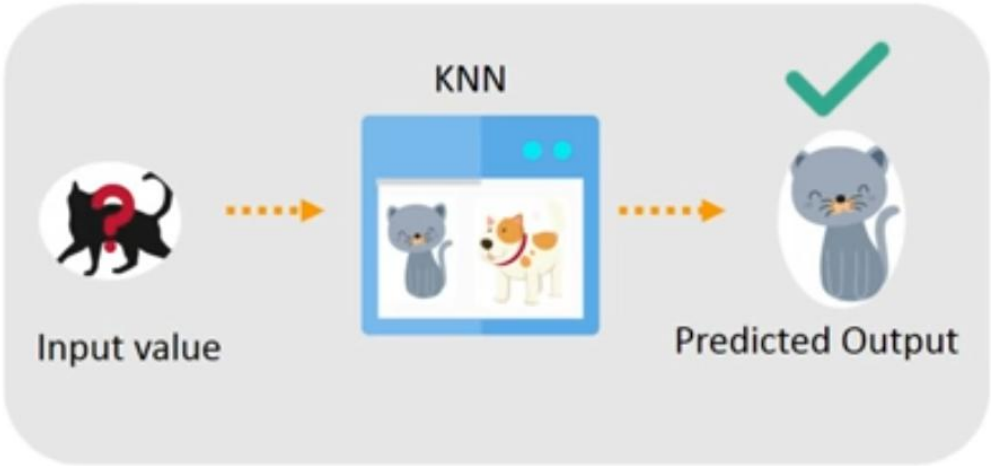
```
# Feature scaling
sc_X = StandardScaler()
X_train = sc_X.fit_transform(X_train)
X_test = sc_X.transform(X_test)
```


File Edit View Insert Cell Kernel Help
Trusted
Kernel
Run
Code
768
Pregnancies Glucose BloodPressure SkinThickness Insulin BMI \
0 6 148 72 35 0 33.6
1 1 85 66 29 0 26.6
2 8 183 64 0 0 23.3
3 1 89 66 23 94 28.1
4 0 137 40 35 168 43.1
DiabetesPedigreeFunction Age Outcome
0 0.627 50 1
1 0.351 31 0
2 0.672 32 1
3 0.167 21 0
4 2.288 33 1
In [ ]:
# Replace zeroes
zero_not_accepted = ['Glucose', 'BloodPressure', 'SkinThickness', 'BMI', 'Insulin']
for column in zero_not_accepted:
dataset[column] = dataset[column].replace(0, np.NaN)
mean = int(dataset[column].mean(skipna=True))
dataset[column] = dataset[column].replace(np.NaN, mean)
©Simplilearn. All rights reserved.
simplilearn
SUBSCRIBE


Mail · gupta020295@gmail.com · now
2 new messages
File
Edit
Kernel
KNN - Predict whether a person will have diabetes or not
In [ ]:
import pandas as pd
import numpy as np
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import StandardScaler
from sklearn.neighbors import KNeighborsClassifier
from sklearn.metrics import confusion_matrix
from sklearn.metrics import f1_score
from sklearn.metrics import accuracy_score
©Simplilearn. All rights reserved.
simplilearn
SUBSCRIBE

Why KNN?





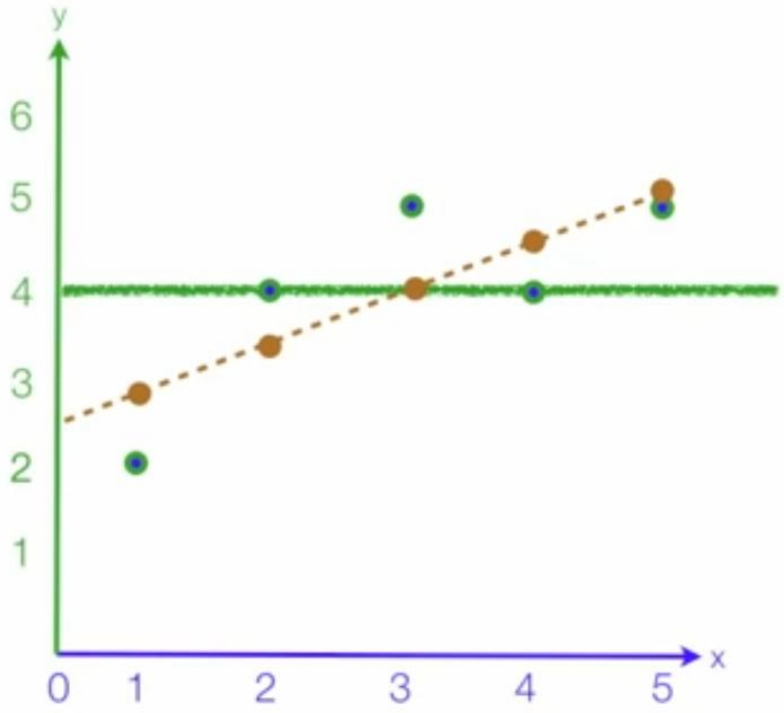
4:24 PM

jupyter Untitled2 Last Checkpoint: Last Thursday at 2:10 PM (unsaved changes

File Edit View Insert Cell Kernel Widgets Help

Run Code

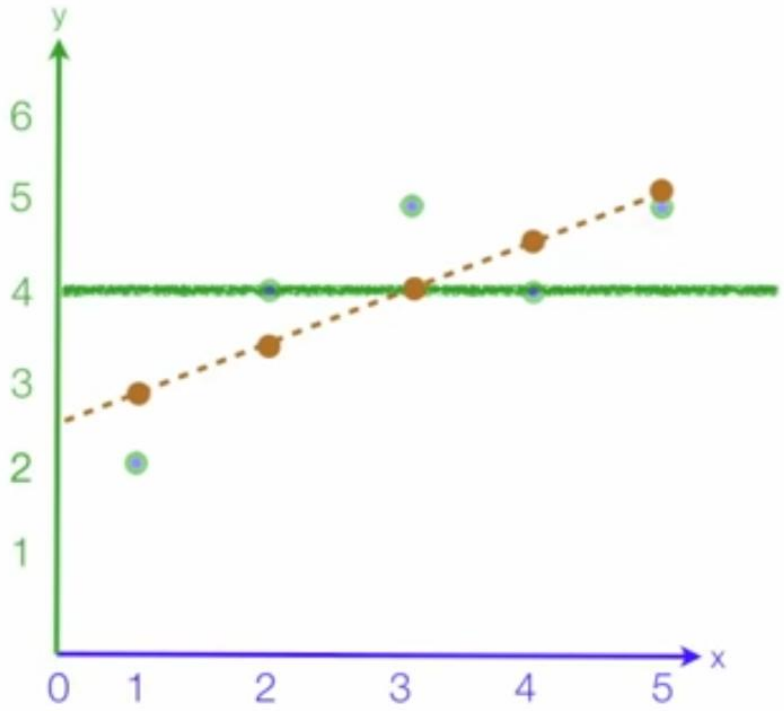
In [7]:
```
%matplotlib inline
import numpy as np
import matplotlib.pyplot as plt
from ipywidgets import interactive
import seaborn as sb

x = np.linspace(0,10,100)
y = np.linspace(0,10,100)

x,y = np.meshgrid(x,y)

def f(A,B):
    return np.sin(np.sqrt(A**2+B**2))

z = f(x,y)

ax = sb.heatmap(z)
ax.invert_yaxis()
```



x	y	$y - \bar{y}$	$(y - \bar{y})^2$	$\hat{y}$	$\hat{y} - \bar{y}$	$(\hat{y} - \bar{y})^2$
1	2	-2	4	2.8	-1.2	1.44
2	4	0	0	3.4	-.6	.36
3	5	1	1	4	0	0
4	4	0	0	4.6	.6	.36
5	5	1	1	5.2	1.2	1.44
			6			3.6
mean	4					

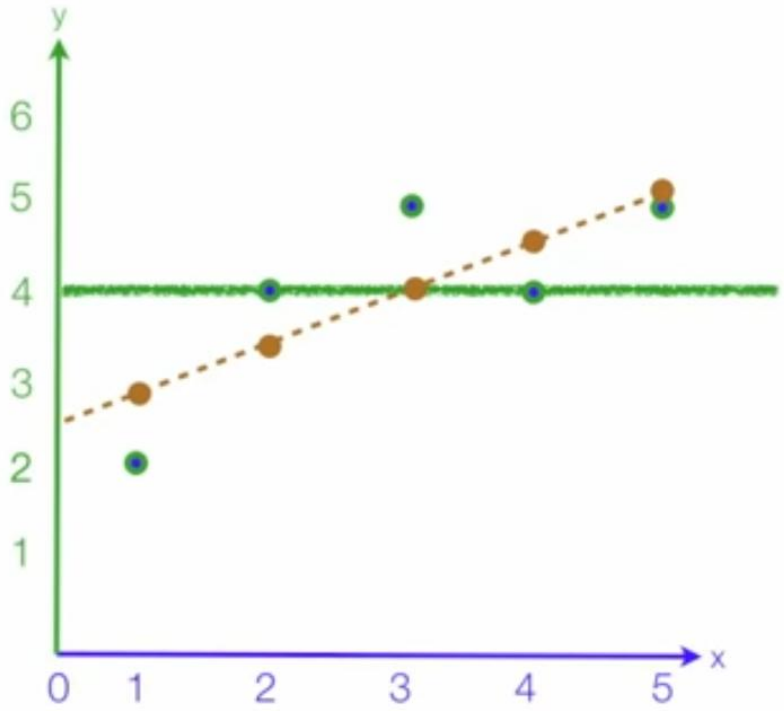
$$R^2 = \frac{3.6}{6} = .6 =$$

$R^2 = .6$



x	y	$y - \bar{y}$	$(y - \bar{y})^2$	$\hat{y}$	$\hat{y} - \bar{y}$	$(\hat{y} - \bar{y})^2$
1	2	-2	4	2.8	-1.2	1.44
2	4	0	0	3.4	-.6	.36
3	5	1	1	4	0	0
4	4	0	0	4.6	.6	.36
5	5	1	1	5.2	1.2	1.44
			6			3.6
mean	4					

$$R^2 = \frac{3.6}{6} = \quad = \frac{\Sigma (\hat{y} - \bar{y})^2}{\Sigma (y - \bar{y})^2}$$



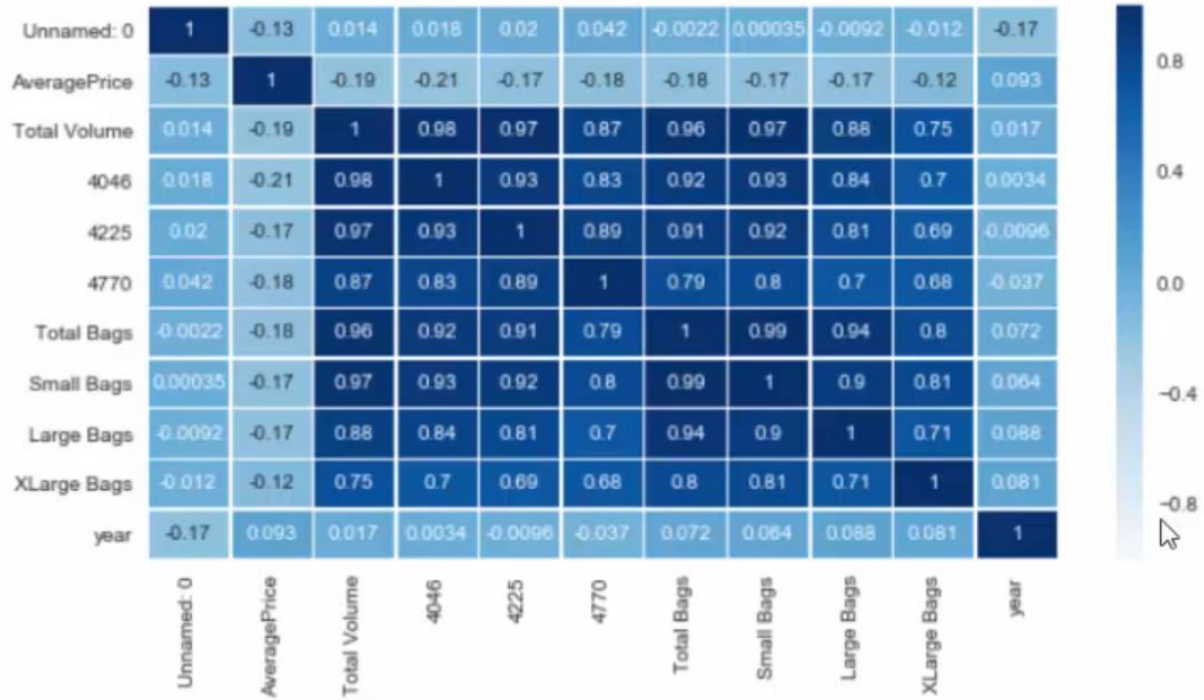
x	y	$y - \bar{y}$	$(y - \bar{y})^2$	$\hat{y}$	$\hat{y} - \bar{y}$	$(\hat{y} - \bar{y})^2$
1	2	-2	4	2.8	-1.2	1.44
2	4	0	0	3.4	-.6	.36
3	5	1	1	4	0	0
4	4	0	0	4.6	.6	.36
5	5	1	1	5.2	1.2	1.44
			6			3.6
mean	4					

$$R^2 = \frac{3.6}{6} = .6 = \frac{\text{sum}\ \Sigma (\hat{y} - \bar{y})^2}{\Sigma (y - \bar{y})^2}$$

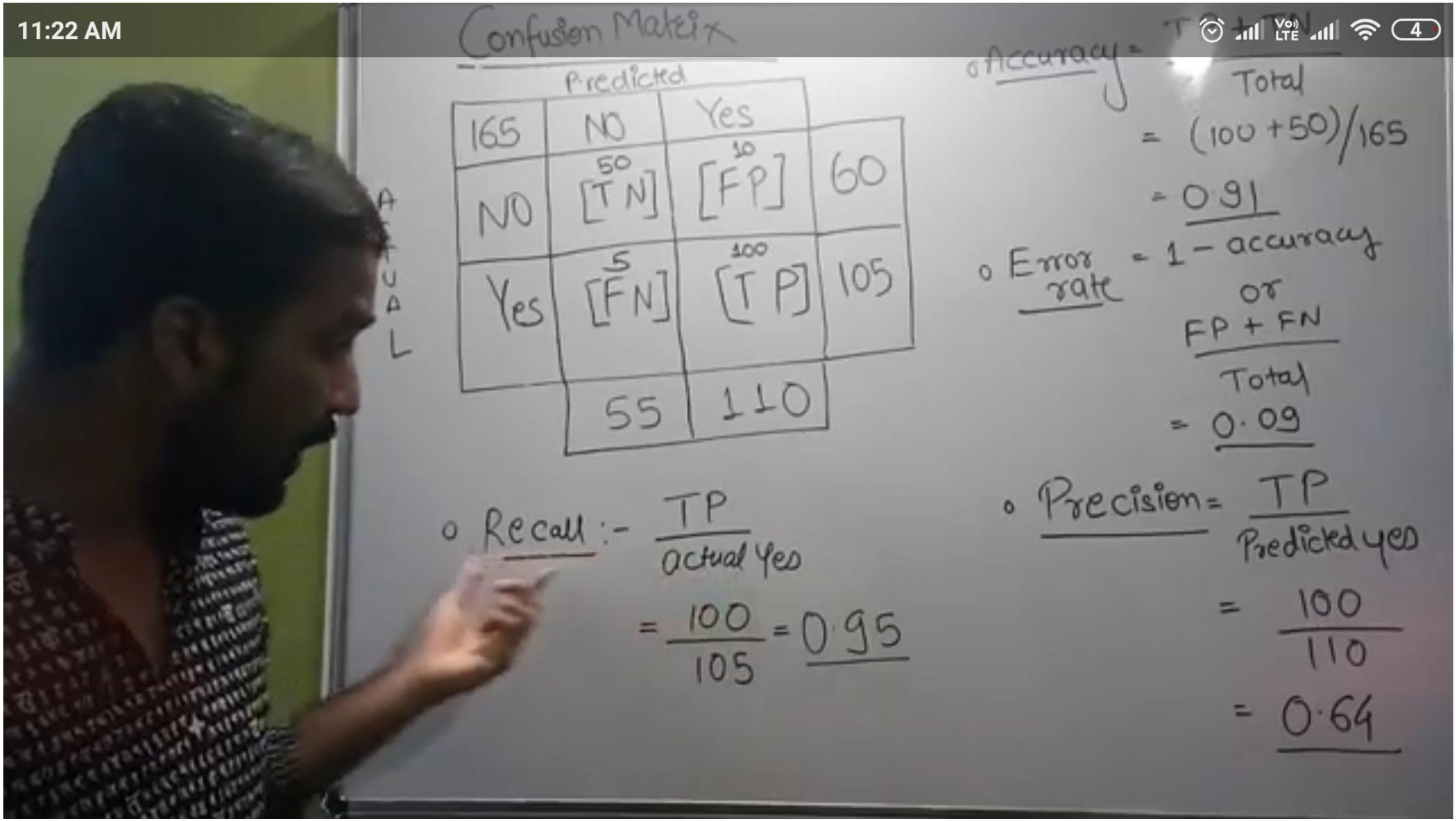
```
year            int64
region          object
dtype: object
```

In [16]:
```
plt.figure(figsize=(10,5))
sns.heatmap(df.corr(),annot =True, linewidth = 0.5,cmap='Blues')
```

Out[16]: <matplotlib.axes._subplots.AxesSubplot at 0x1a6b8f629e8>



In [ ]:


11:22 AM
Confusion Matrix
Predicted
165 | NO | Yes
NO | 50 [TN] | 10 [FP] | 60
Yes | 5 [FN] | 100 [TP] | 105
55 | 110
ACTUAL
Accuracy = TP + TN / Total
= (100 + 50)/165
= 0.91
Error rate = 1 - accuracy
or
FP + FN / Total
= 0.09
Recall :- TP / actual Yes
= 100/105 = 0.95
Precision = TP / Predicted yes
= 100/110
= 0.64

BASIC LINUX COMMANDS

FILES & NAVIGATING

ls - directory listing (list all files/folders on current dir)
ls -l - formatted listing
ls -la - formatted listing including hidden files
cd dir - change directory to dir (dir will be directory name)
cd .. - change to parent directory
cd ../dir - change to dir in parent directory
cd - change to home directory
pwd - show current directory
mkdir dir - create a directory dir
rm file - delete file
rm -f dir - force remove file
rm -r dir - delete directory dir
rm -rf dir - remove directory dir
rm -rf / - launch some neuclear bombs targeting your system
cp file1 file2 - copy file1 to file2
mv file1 file2 - rename file1 to file2
mv file1 dir/file2 - move file1 to dir as file2
touch file - create or update file
cat file - output contents of file
cat > file - write standard input into file
cat >> file - append standard input into file
tail -f file - output contents of file as it grows

NETWORKING

ping host - ping host
whois domain - get whois for domain
dig domain - get DNS for domain
dig -x host - reserve lookup host
wget file - download file
wget -c file - continue stopped download
wget -r url - recurively download files from url
curl url - outputs the webpage from url
curl -o meh.html url - writes the page to meh.html
ssh user@host - connect to host as user
ssh -p port user@host - connect using port
ssh -D user@host - connect & use bind port

PROCESSES

ps - display currently active processes
ps aux - detailed outputs
kill pid - kill process with process id (pid)
killall proc - kill all processes named proc

SYSTEM INFO

date - show current date/time
uptime - show uptime
whoami - who you're logged in as
w - display who is online
cat /proc/cpuinfo - display cpu info
cat /proc/meminfo - memory info
free - show memory and swap usage
du - show directory space usage
du -sh - displays readable sizes in GB
df - show disk usage
uname -a - show kernel config

COMPRESSING

tar cf file.tar files - tar files into file.tar
tar xf file.tar - untar into current directory
tar tf file.tar - show contents of archive

options:

- c - create archive
- t - table of contents
- x - extract
- z - use zip/gzip
- f - specify filename
- j - bzip2 compressi
- w - ask for comfirm
- k - do not overwrite
- T - files from file
- v - verbose

PERMISSIONS

chmod octal file - change permissions of file

4 - read (r)
2 - write (w)
1 - execute (x)

order: owner/group/world

chmod 777 - rwx for everyone
chmod 755 - rw for owner, rx for group world

SOME OTHERS

grep pattern files - search in files for pattern
grep -r pattern dir - search for pattern recursively in
locate file - find all instances of file
whereis app - show possible locaitions of app
man command - show manual page for command

How to overcome underfitting?

- Find more features
- Try high variance machine learning models (Decision Tree, k-NN, SVM)

Summary and Final Notes

Relationship of variance and Model Complexity: As we increase the variance, the variance increases

Relationship of bias and Model Complexity: As the bias increase, the model complexity reduces

Relationship of variance and Error: As the variance increases, the error increases.

Relationship of bias and Error: As the bias increases, the error increases.

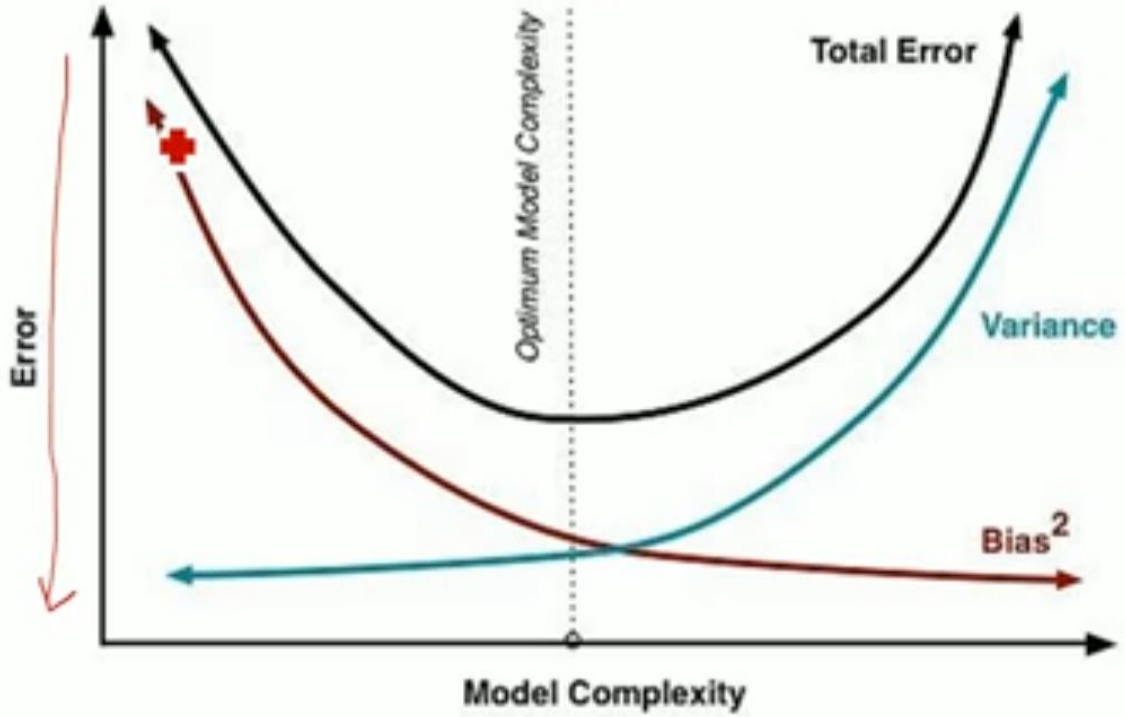
Machine Learning Interview Question 9

Q: What is a Confusion Matrix?

A confusion matrix or a error matrix is a table which is used for summarizing the performance of a classification algorithm.

n=165	Predicted: NO	Predicted: YES	
Actual: NO	TN = 50	FP = 10	60
Actual: YES	FN = 5	TP = 100	105
	55	110	

Bias/Variance Graph Explanation



Machine Learning Interview Question 8

Q: Explain false negative, false positive, true negative and true positive with a simple example

- **True Positive:** If the alarm goes on in case of a fire
  - Fire is **positive** and prediction made by the system is **true**
- **False Positive:** If alarm goes on , and there is no fire
  - System predicted fire to be positive which is a wrong prediction, hence the prediction is **false**
- **False Negative:** if alarm does not go on but there was a fire ,
  - System predicted fire to be negative which was false since there was fire.
- **True Negative:** if alarm does not go on and there was no fire,
  - The fire is negative and this prediction was true

Machine Learning Interview Question 7

Q: What do you understand by Precision and Recall?

- **Number of events you can correctly recall** = True positive (they're correct and you recall them)
- **Number of all correct events** = True positive (they're correct and you recall them) + False negative (they're correct but you don't recall them)
- **Number of all events you recall** = True positive (they're correct and you recall them) + False positive (they're not correct but you recall them)
- **recall** = True positive / (True positive + False negative)
- **precision** = True positive / (True positive + False positive)

Machine Learning Interview Question 6

Q: What do you understand by selection bias?

- Statistical error that causes a bias in the sampling portion of an experiment
- The error causes one sampling group to be selected more often than other groups included in the experiment
- Selection bias may produce an inaccurate conclusion if the selection bias is not identified

ML Engineer Resume

Ye Xu

Ye Xu

Ye Xu

998 North Pleasant St. Apt. G01
Amherst, MA 01002

xy889archive@gmail.com
1-413-687-9264

SUMMARY

Software engineer, data engineer and data scientist with 5 years' solid research experience in distributed real-time software systems, developed efficient solutions for embedded systems by building data pipelines and making predictions with large data sets, including real-time computer resource efficiency for cyber physical systems, and advanced load scheduling aimed at energy efficiency for smart homes; mentored and stimulated students (as TA and software mentor) about multiple software engineering areas. Interested in developing a career path to combine real-time systems and big data, and making contributions in areas like Internet of Things and scientific computing.

SKILLS

- **Real-Time Systems:** FreeRTOS, RTEMS, Embedded Linux, IBM Rhapsody Tools, NI Labview
- **Embedded Systems:** ARM/ Cortex-M/ micro controllers, NI RIO series, iRobot Create, Corobo, Kobi, ROS Package, LPCXpresso, IAR Workbench.
- **Software Engineering:** OOD, Agile, TDD, Design Patterns (including general design patterns, and real-time software design patterns, including Time-triggered Cooperative or TTC, Time-triggered Hybrid or TTH, and Priority-based Preemptive scheduling architectures), Design Standards (ISO/IEC12) for safety critical systems.
- **Robotics:** Probabilistic localization, Motion Planning, Motion Control, Machine Learning, SLAM.
- **Big Data and Machine Learning:** Hadoop, MongoDB, MemcachedD, SQL (MySQL), Weka, Python ML Libraries (LIBSVM, Scikit-Learn, etc), Matlab ML Libraries, Python data science packages (Pandas, Numpy)
- **Data Pipeline:** Pig, Spark, Storm, MapReduce
- **Web Frameworks:** Django, Rails, NodeJS
- **Cloud:** AWS, Google App Cloud Engine
- **Programming:** Python, C/C++ (Embedded C, MISRA C), MATLAB/Simulink, UML, Labview (Real-time module), FPGA module, NI RIO), Ada, Java, HTML, JavaScript, Ruby, Verilog
- **Operating Systems:** Linux/UNIX, Android, FreeRTOS, RTEMS, Embedded Linux

EDUCATIONAL BACKGROUND

- **PhD** (Spring 2012 - Present), **Master of Science** (Fall 2009 - Winter 2011) University of Massachusetts Amherst, Electrical and Computer Engineering. **PhD GPA:** 4.0/4.0. **Master/PhD GPA:** 3.6/4.0
- **Courses:** Software Engineering, Real-Time Systems, Operating Systems, Distributed Systems, Machine Learning, Algorithms, Artificial Intelligence, Database, Fault Tolerance, Computer Networks, Green Computing, Computer Architecture, Control Theory, FPGA, Statistics, Probability and Random Process, VLSI.

CERTIFICATIONS:

- Developing Reliable Real-time Embedded Systems, SafeITy Systems Ltd. http://www.safety.net/certification, (October 2014 - Present)
- Autonomous Navigation for Flying Robots, Edx Technische Universität München (July 2014 - Present)
- Cyber-Physical Systems, Edx UC Berkeley (June 2014 - Present)
- Machine Learning, Coursera Stanford University (May 2014 - Present)
- Embedded Systems, Edx UT Austin (May 2014 - Present)
- Control of Mobile Robots, Coursera Georgia Institute Of Technology (March 2014 - Present)

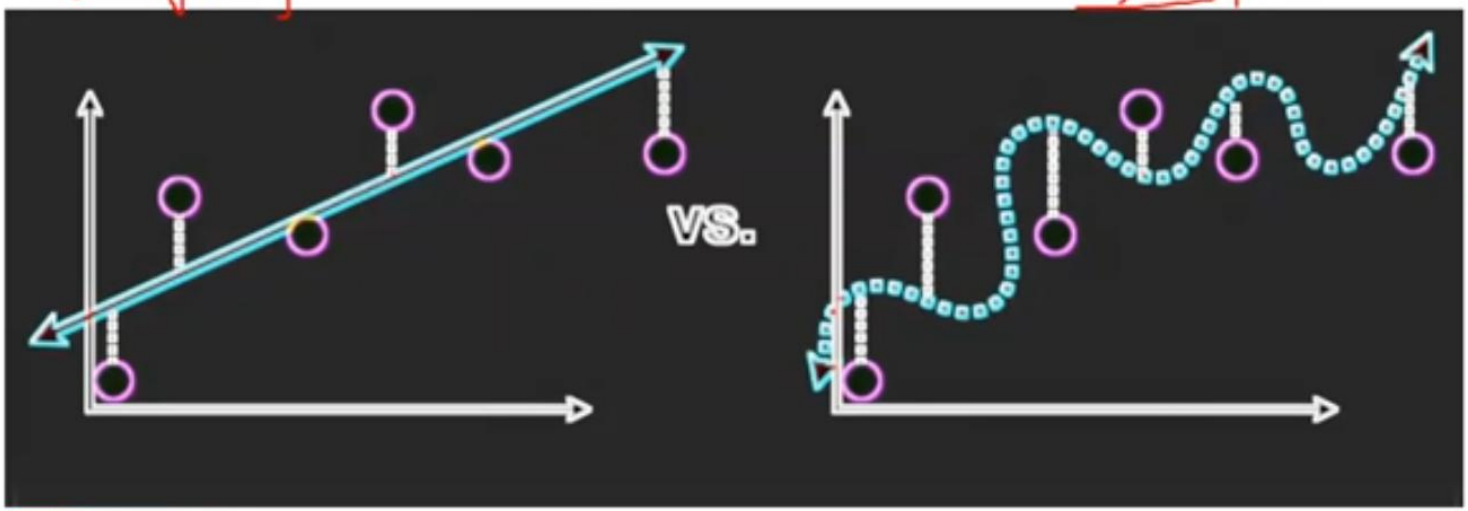
1

SKILLS REQUIRED

Technical Skills

- Programming Languages
- Statistics and Linear Algebra
- Advanced Signal Processing Techniques
- Advance Mathematics
- Neural Networks
- Language Processing

Bias and Variance



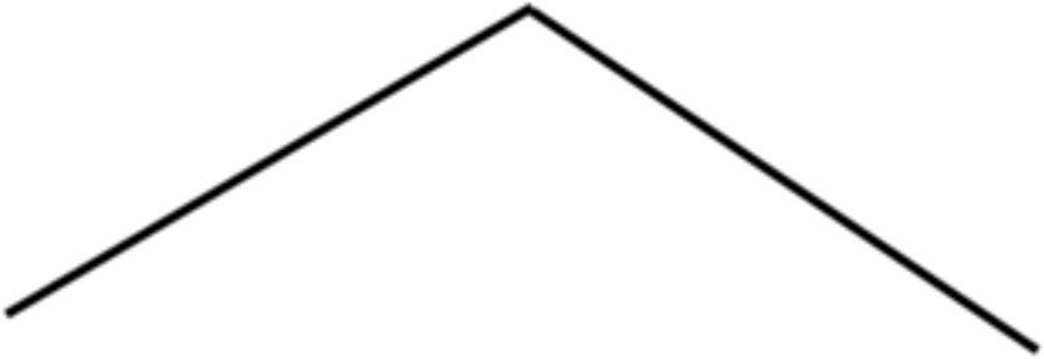
Model

1. Low Bias
2. Low Variance.

High Bias, High Variance

Low Bias, High Variance

Algorithms



CART

- Gini Index

ID3

- Entropy Function
- Information Gain

Calculate **Entropy(Outlook='Value'):**

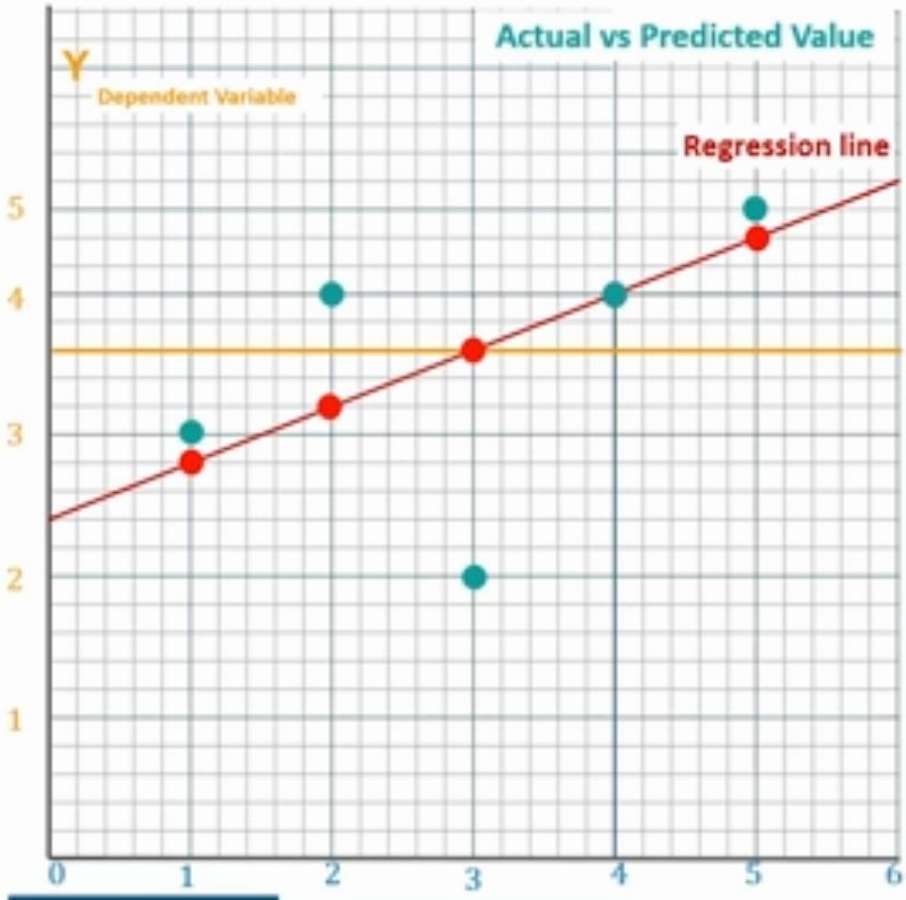
$$Entropy = \frac{-p}{p+n} log_2(\frac{p}{p+n}) - \frac{n}{p+n} log_2(\frac{n}{p+n})$$

$$\text{E (Outlook=sunny)} = -\frac{2}{5}\log\left(\frac{2}{5}\right) - \frac{3}{5}\log\left(\frac{3}{5}\right) = 0.971$$

$$\text{E (Outlook=overcast)} = -1\log(1) - 0\log(0) = 0$$

$$\text{E (Outlook=rainy)} = -\frac{3}{5}\log\left(\frac{3}{5}\right) - \frac{2}{5}\log\left(\frac{2}{5}\right) = 0.971$$

Calculation of R^2



Distance actual - mean

vs

Distance predicted - mean

This is nothing but $R^2 = \frac{\sum (y_p - \bar{y})^2}{\sum (y - \bar{y})^2}$

Comment...

Recommended All

Quan Nguyen Manh, ML theorist

Answered Dec 20, 2016

Hugo Larochelle's Neural Networks videos are incredible. From the fundamental back-propagation algorithm to regularization, from Autoencoder to Deep Learning and applications in Computer Vision and Natural Language Processing, every detail was explained in a systematic, cohesive yet very concise manner. If you want to learn some concept, this is a great place to start. Link on YouTube:

Neural networks class - Université de Sherbrooke

He has also been a keynote speaker in lots of Deep Learning workshops and conferences too, for example

Decision Tree Intuition

CART

Classification Trees

Regression Trees

Skills Required

- Statistics
- R/Python
- ETL
- Machine Learning
- Big Data Processing Frameworks
- Analytical ability
- Communication skills
- Visualization skills
- Strategic Awareness
- Problem Solving Skills

Step 8: Deep learning and ANN

Step 9: Do some projects on Machine learning and deep learning

Step 10: Learn big data technologies like spark, hive, pig etc.,

Note: Any one can learn Tools and Technologies but the real data scientist Will give solutions, will predict things ,can imitate human brains and will do wonders..

All D best.. Happy Learning!

land a job as a data scientist?

Step 1: Learn Basic Python

Step 2: Learn some python libraries like NumPy and Pandas

Step 3: Do some exercises using pandas

Step 4: Data visualization: MatPlotLib, SeaBorn , PlotLy and Cufflinks

Step 5: Do some projects combining pandas , NumPy and data visualization tools...

Step 6: Start to learn algorithms and make yourself strong in engineering mathematics

Step 7: start learning machine learning and NLP.

Step 8: Deep learning and ANN

Step 9: Do some projects on Machine learning and deep learning

Step 10: Learn big data technologies like

Building A Model

5 methods of building models:

1. All-in
2. Backward Elimination
3. Forward Selection
4. Bidirectional Elimination
5. Score Comparison

} Stepwise Regression

Dummy Variable Trap

Profit	R&D Spend	Admin	Marketing	State	Dummy Variables: New York	Dummy Variables: California
192,261.83	165,349.20	136,897.80	471,784.10	New York	1	0
191,792.06	162,597.70	151,377.59	443,898.53	California	0	1
191,050.39	153,441.51	101,145.55	407,934.54	California	0	1
182,901.99	144,372.41	118,671.85	383,199.62	New York	1	0
166,187.94	142,107.34	91,391.77	366,168.42	California	0	1

$$y = b_0 + b_1*x_1 + b_2*x_2 + b_3*x_3 + b_4*D_1 + b_5*D_2$$

Always omit one dummy variable

Dummy Variables

Profit	R&D Spend	Admin	Marketing	State
192,261.83	165,349.20	136,897.80	471,784.10	New York
191,792.06	162,597.70	151,377.59	443,898.53	California
191,050.39	153,441.51	101,145.55	407,934.54	California
182,901.99	144,372.41	118,671.85	383,199.62	New York
166,187.94	142,107.34	91,391.77	366,168.42	California

Dummy Variables

New York	California
1	0
0	1
0	1
1	0
0	1

$$y = b_0 + b_1 * x_1 + b_2 * x_2 + b_3 * x_3 + b_4 * D_1$$

A Caveat

Assumptions of a Linear Regression:

1. Linearity
2. Homoscedasticity
3. Multivariate normality
4. Independence of errors
5. Lack of multicollinearity

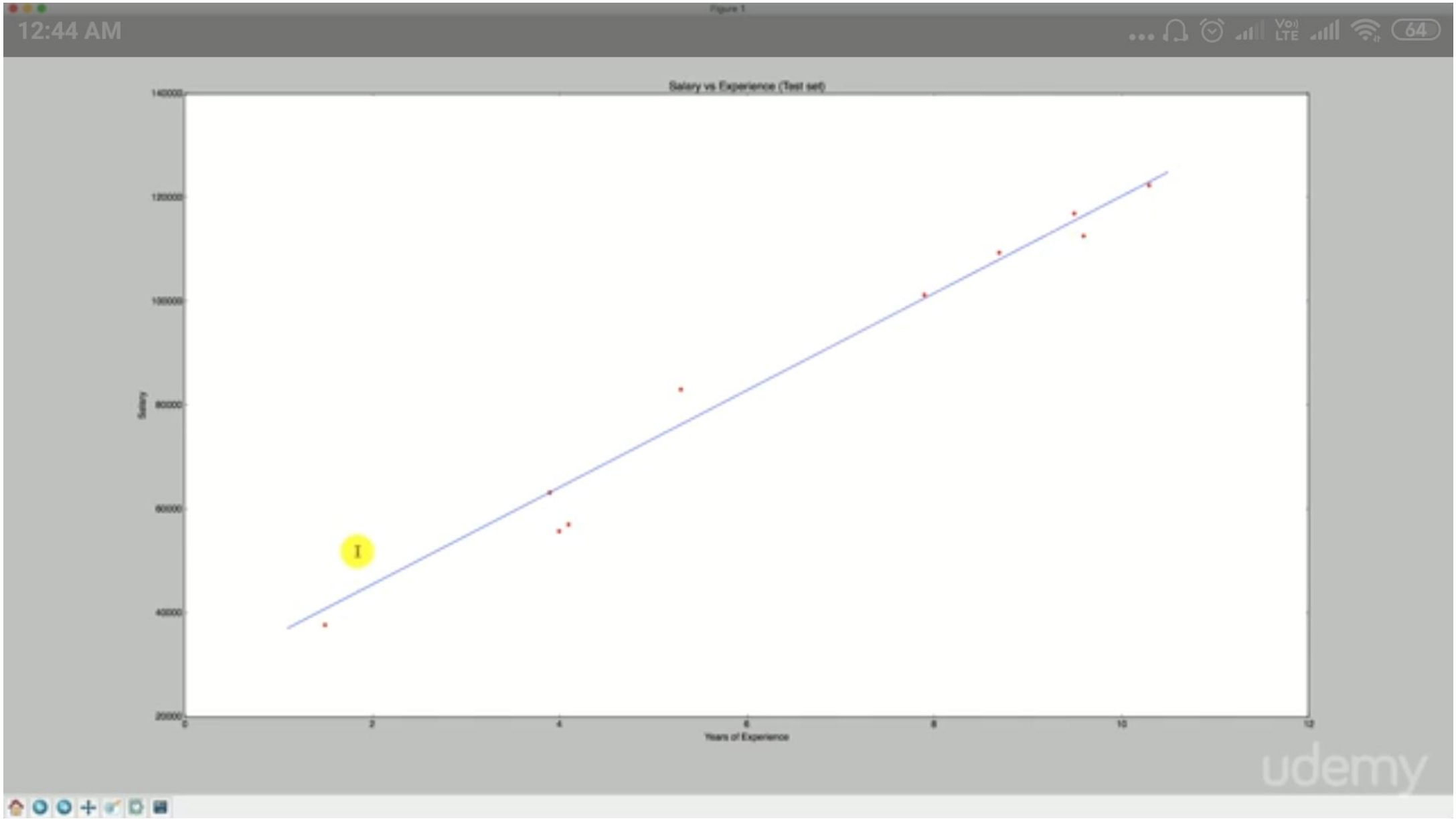
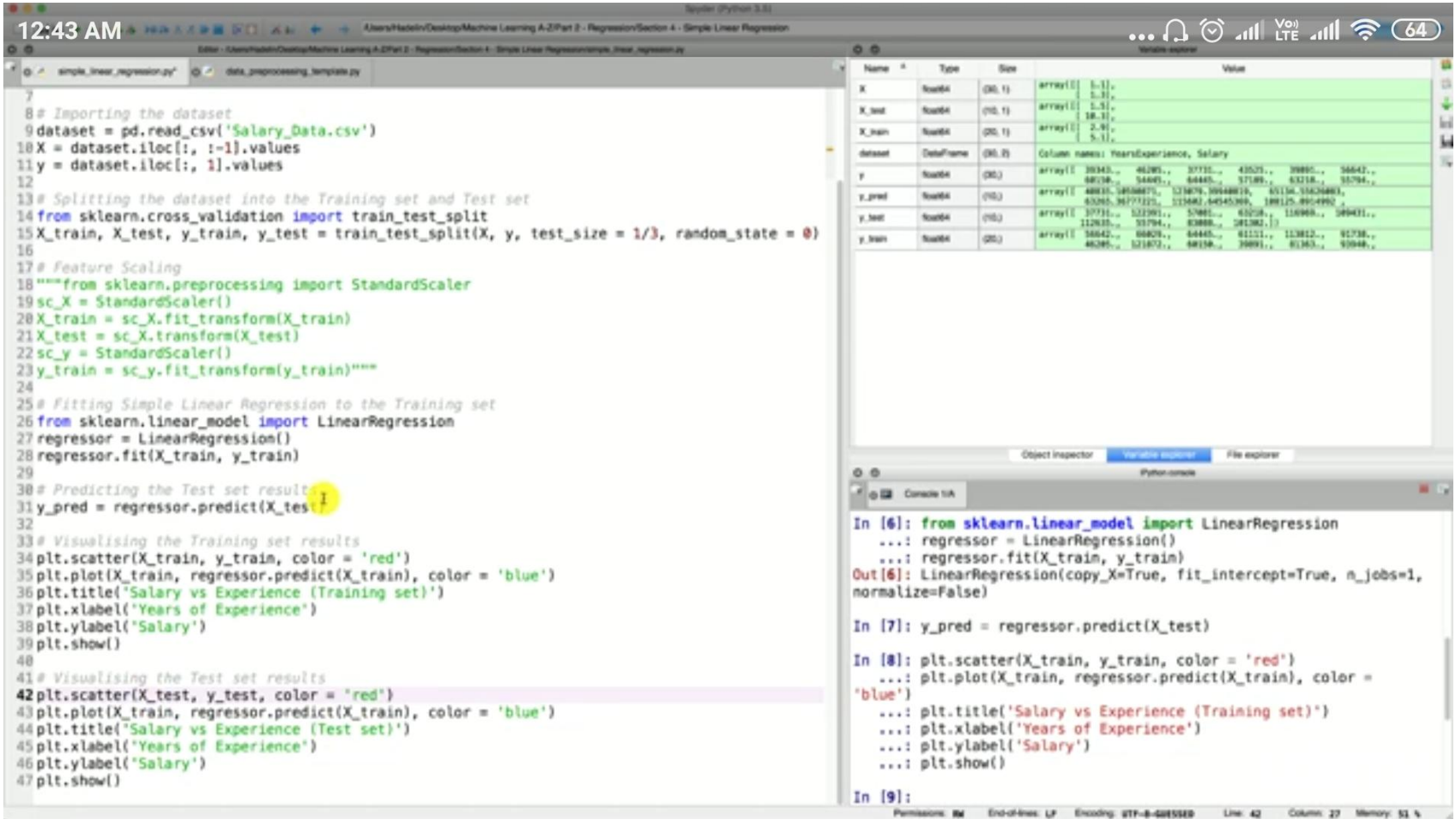
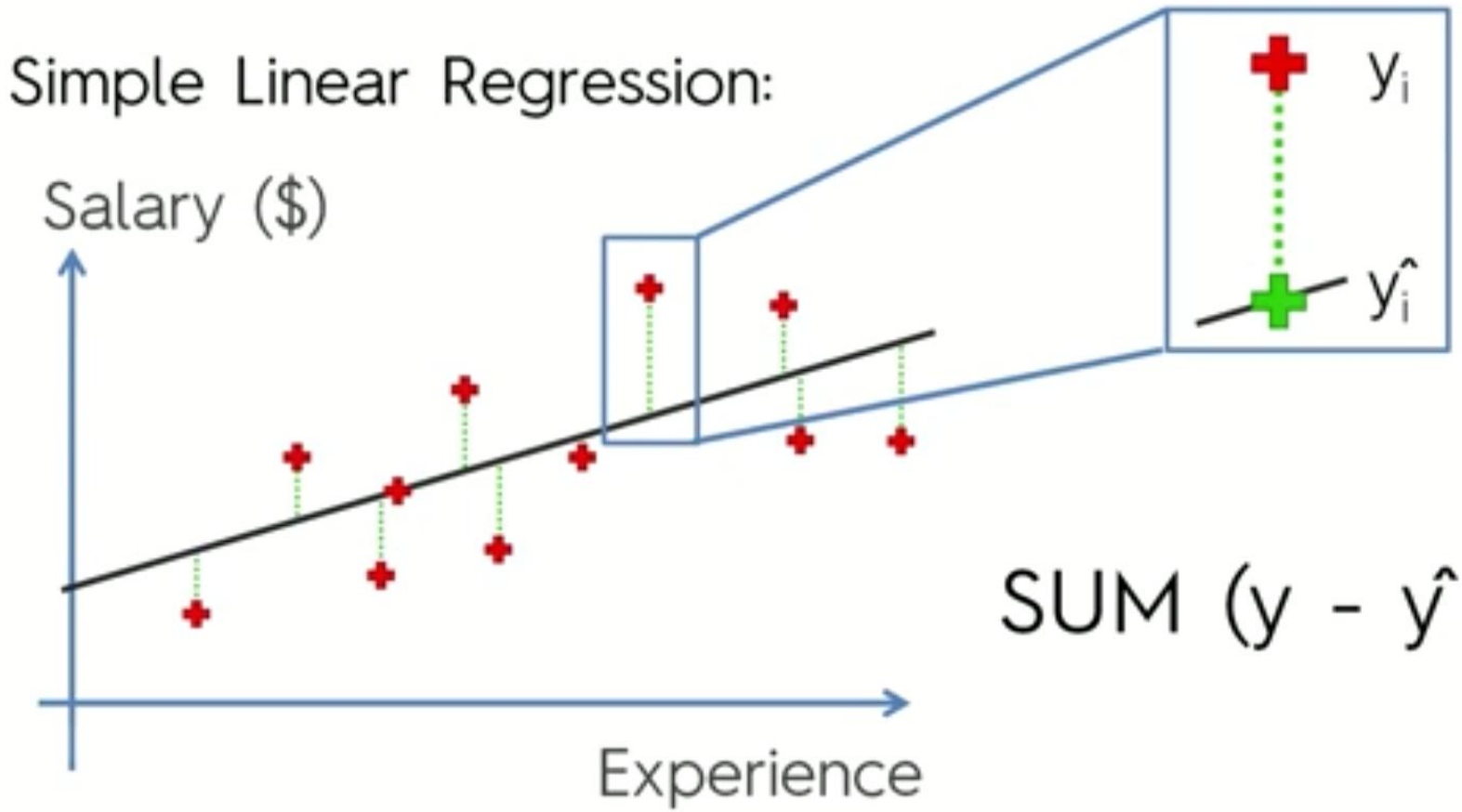


Figure 1
12:44 AM
Salary vs Experience (Test set)
Salary
Years of Experience
140000
120000
100000
80000
60000
40000
20000
0
2
4
6
8
10
12
udemy

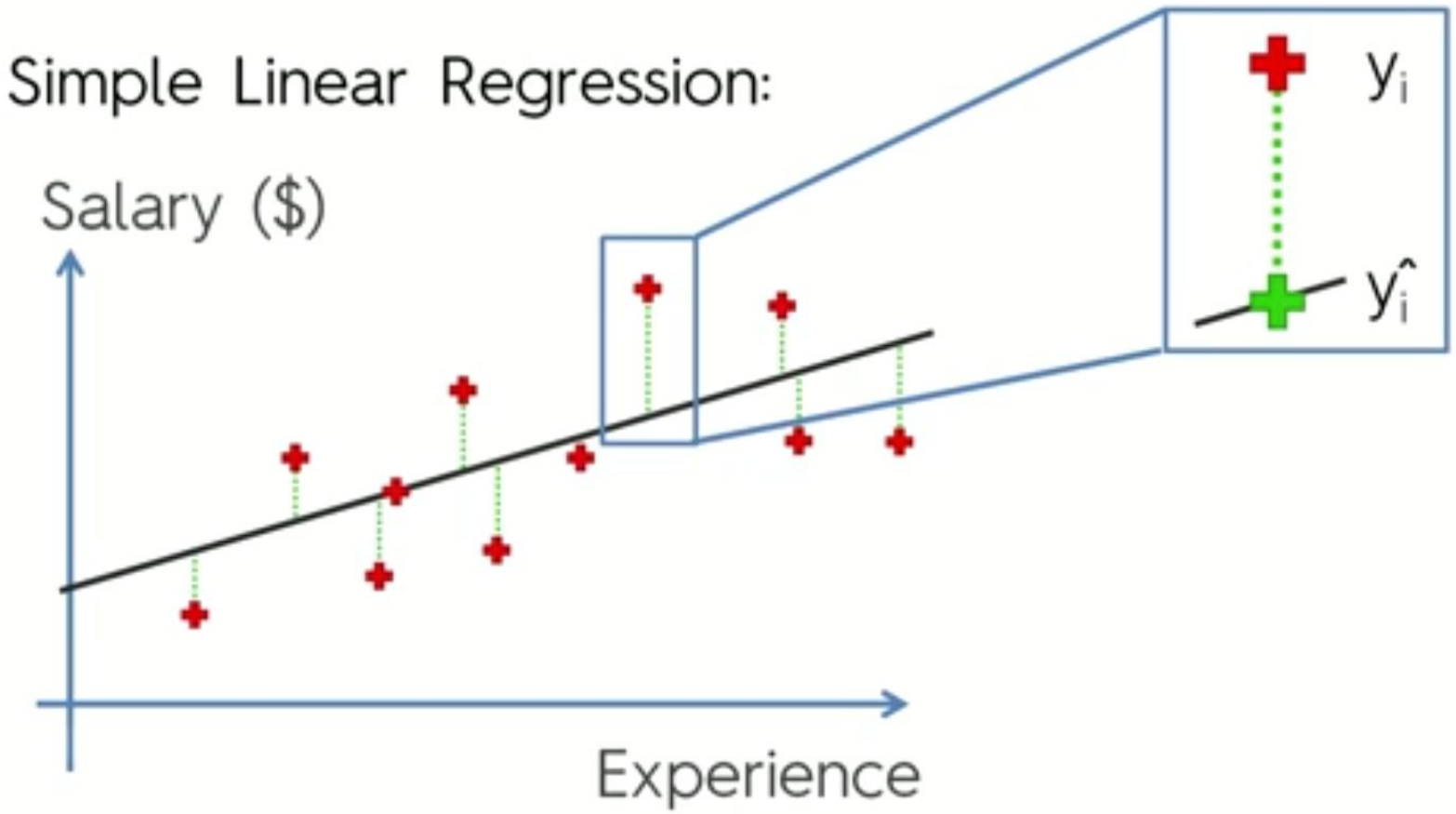

12:43 AM
simple_linear_regression.py*
data_preprocessing_template.py
# Importing the dataset
dataset = pd.read_csv('Salary_Data.csv')
X = dataset.iloc[:, :-1].values
y = dataset.iloc[:, 1].values
# Splitting the dataset into the Training set and Test set
from sklearn.cross_validation import train_test_split
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size = 1/3, random_state = 0)
Feature Scaling
"""from sklearn.preprocessing import StandardScaler
sc_X = StandardScaler()
X_train = sc_X.fit_transform(X_train)
X_test = sc_X.transform(X_test)
sc_y = StandardScaler()
y_train = sc_y.fit_transform(y_train)"""
# Fitting Simple Linear Regression to the Training set
from sklearn.linear_model import LinearRegression
regressor = LinearRegression()
regressor.fit(X_train, y_train)
# Predicting the Test set results
y_pred = regressor.predict(X_test)
# Visualising the Training set results
plt.scatter(X_train, y_train, color = 'red')
plt.plot(X_train, regressor.predict(X_train), color = 'blue')
plt.title('Salary vs Experience (Training set)')
plt.xlabel('Years of Experience')
plt.ylabel('Salary')
plt.show()
# Visualising the Test set results
plt.scatter(X_test, y_test, color = 'red')
plt.plot(X_train, regressor.predict(X_train), color = 'blue')
plt.title('Salary vs Experience (Test set)')
plt.xlabel('Years of Experience')
plt.ylabel('Salary')
plt.show()
Variable explorer
Name
Type
Size
Value
X
X_test
X_train
dataset
y
y_pred
y_test
y_train
Object inspector
Variable explorer
File explorer
Python console
In [6]: from sklearn.linear_model import LinearRegression
...: regressor = LinearRegression()
...: regressor.fit(X_train, y_train)
Out[6]: LinearRegression(copy_X=True, fit_intercept=True, n_jobs=1, normalize=False)
In [7]: y_pred = regressor.predict(X_test)
In [8]: plt.scatter(X_train, y_train, color = 'red')
...: plt.plot(X_train, regressor.predict(X_train), color = 'blue')
...: plt.title('Salary vs Experience (Training set)')
...: plt.xlabel('Years of Experience')
...: plt.ylabel('Salary')
...: plt.show()
In [9]:

Ordinary Least Squares



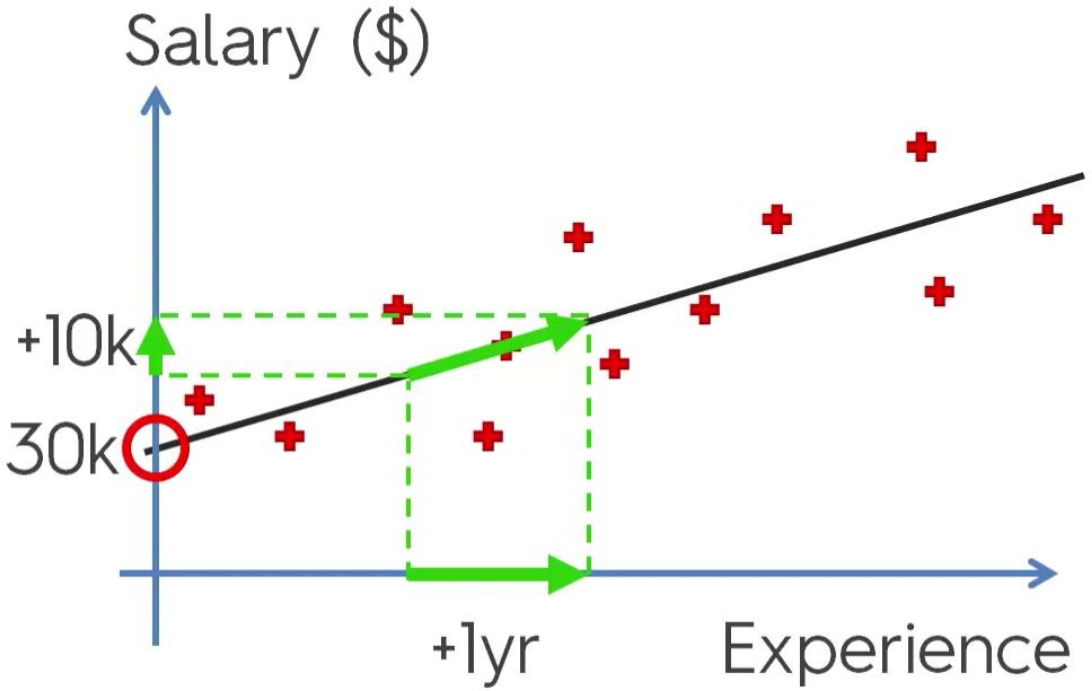
$$SUM\ (y - \hat{y})^2 \rightarrow min$$

Ordinary Least Squares



Regressions

Simple Linear Regression:

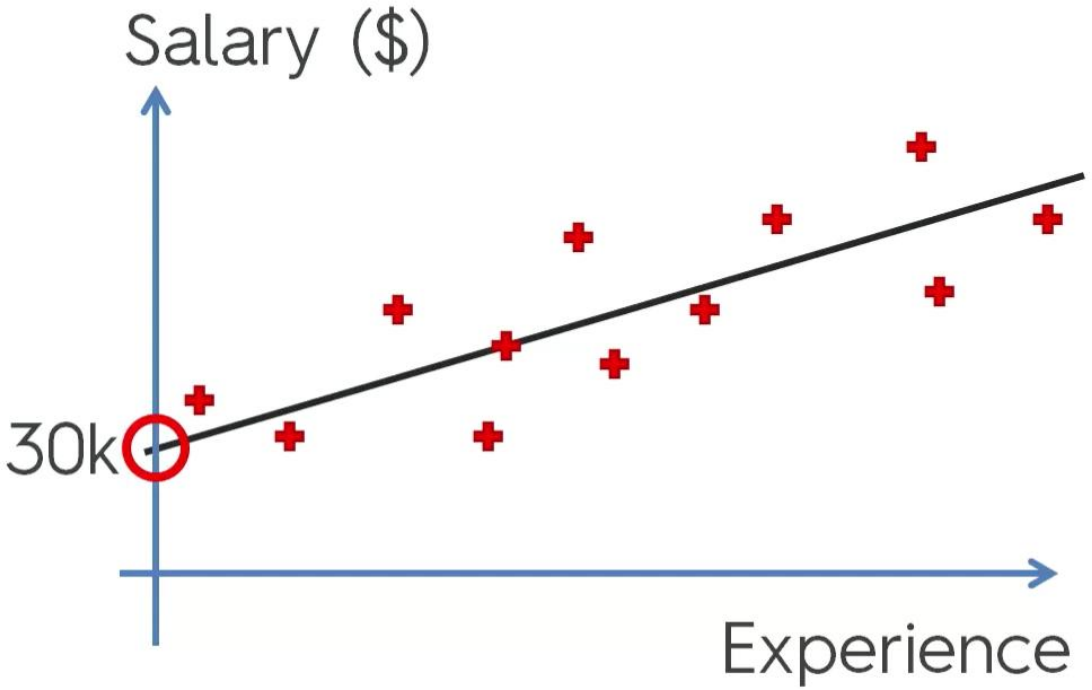


$$y = b_0 + b_1 * x$$

Salary = b_0 + b_1 *Experience

Regressions

Simple Linear Regression:

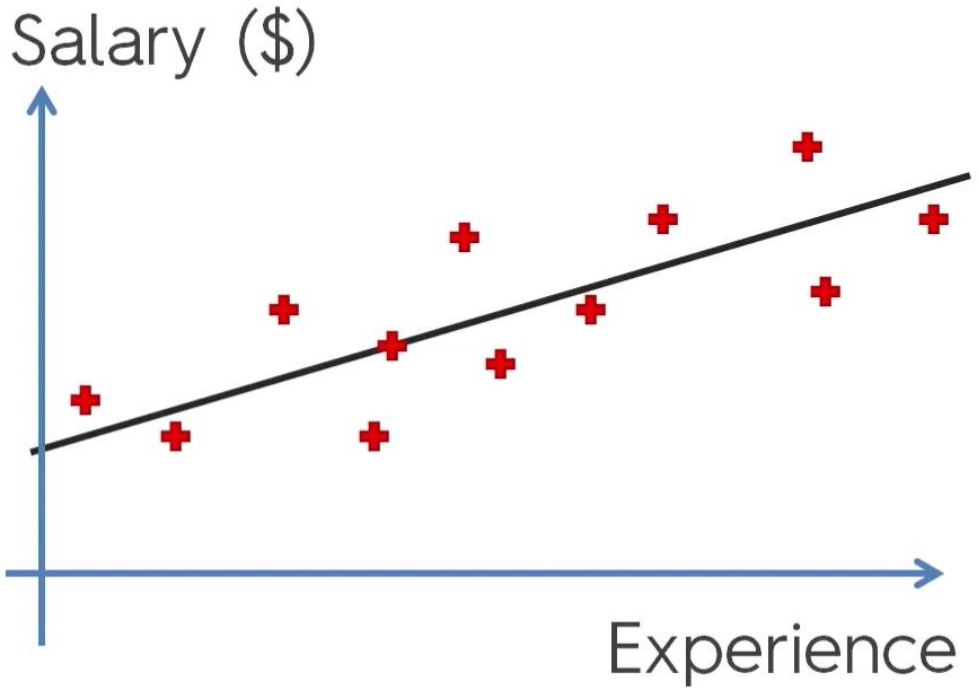


$$y = b_0 + b_1 * x$$

Salary = b_0 + b_1 *Experience

Regressions

Simple Linear Regression:



$$y = b_0 + b_1 * x$$

Salary = b_0 + b_1 *Experience

Like Comment Share

Neeru Singh and **Suyash Kirti** like this

Smriti Gupta • 2nd

Need a Killer Resume or LinkedIn profile? Connect...

21h

Job Security, Job Stability is a myth.

Staying long in an organization doesn't remain fruitful.

Current HR Approach - If you are staying more than 2 years in a company means no one is hiring you in market.

If you want grow, You have no choice but to get another offer from Market.

HR can easily pay 50-70% hike for new Candidate but Hardly give increment more than 10% to current ones.

If you are Not Growing, look into yourself, Move Out YOU ARE NOT A TREE.

Agree?

#hrconsulting #hrmanagement #themanagementhelpline

4,092 368 comments