

Company name :-Knowledge Solution India

BATCH NO.-11

GROUP-3

PROJECT -2 TOPIC:- INSURANCE COST PREDICTION

GROUP MEMBERS

Payal kumari(Random Forest Regressor)

Debattam Bhattacharjee(Multiple Linear Regressor)

Poornima P(PCA with MLR)

**TABLE OF CONTENT**

ABSTRACT………………………………………………………………………..4

CHAPTER :- 1

INTRODUCTION……………………………………………………………. ….. 5

CHAPTER :- 2

MULTIPLE LINEAR REGRESSOR…………………………………….………...8

2.1 Introduction………………………………………………………………….....8

2.2 Simple linear regressor……………………………………………….……….10

2.3 Linearity……………………………………………………………………….10

2.4 Multivariate Normality………………………………………………………..11

2.5Independence of Error…………………………………………………………11

2.6Lack of Multicolinearity……………………………………………………….12

2.7Dummy Variables……………………………………………………………...12

2.8 Backward Elimination………………………………………………………...13

2.9 Importing the libraries…………………………………………………..…….13

2.10 Encoding categorical data…………………………………………….……..14

2.11 Splitting the dataset into the training set and test set………………..……….15

2.12 Predicting the test set result..…………………………………………….…..16

2.13 Comparing the test set with predicted value…………………………………17

CHAPTER:- 3

RANDOM FOREST REGRESSOR…………………………………………….20

3.1 Introduction…………………………………………………………………...20

3.2 Ensemble learning…………………………………………………………….21

3.3 Decision tree Regressor……………………………………………………….22

3.4 Importing the libraries………………………………………………………..22

3.5 Importing the dataset…………………………………………………………23

3.6 Splitting the data set into training set and test set……………………………25

3.7 Encoding the categorical value……………………………………………......26

3.8 Predicting the result………………………………………………………...…26

3.9 Comparing the real values with predicted values…………………………….27

3.10 Visualizing the random forest regression result…………………………….28

CHAPTER :- 4

PRINCIPAL COMPONENT ANALYSIS……………………………….……….29

4.1 Introduction…………………………………………………………………...29

4.2 PCA linear regressor……..……………………………………………….…..29

4.3 PCA with missing value…...………………………………………………….30

4.4 Format for (MLR with PCA)………………………………………………….30

4.5 Importing libraries…………………………………………….………………31

4.6 Complete algorithm of MLR with PCA………………………………………31

CHAPTER:- 5

CONCLUSION…………………………………………………………………...32

**ABSTRACT**

The objective of this briefing is to present an overview of the machine learning techniques currently in use or in consideration at statistical agencies worldwide. The machine learning field, which can be briefly defined as enabling computers make successful predictions using past experiences, has exhibited an impressive development recently with the help of the rapid increase in the storage capacity and processing power of computers. Together with many other disciplines, machine learning methods have been widely employed in bioinformatics. The difficulties and cost of biological analyses have led to the development of sophisticated machine learning approaches for this application area. In this chapter, we first review the fundamental concepts of machine learning such as feature assessment, unsupervised versus supervised learning and types of classification. Then, we point out the main issues of designing machine learning experiments and their performance evaluation. Finally, we introduce some supervised learning methods. For predicting dataset and building models we used four models of machine learning

MULTIPLE LINEAR REGRESSION

RANDOM FOREST REGRESSION

MLR WITH PCA(PRINCIPAL COMPONENT ANNALYSIS)

RFR WITH PCA

**INTRODUCTION**

In the statistical context, Machine Learning is defined as an application of artificial intelligence where available information is used through algorithms to process or assist the processing of statistical data. While Machine Learning involves concepts of automation, it requires human guidance. Machine Learning involves a high level of generalisation in order to get a system that performs well on yet unseen data instances.

Machine learning is a relatively new discipline within Computer Science that provides a collection of data analysis techniques. Some of these techniques are based on well established statistical methods (e.g. logistic regression and principal component analysis) while many others are not. Most statistical techniques follow the paradigm of determining a particular probabilistic model that best describes observed data among a class of related models. Similarly, most machine learning techniques are designed to find models that best fit data (i.e. they solve certain optimization problems), except that these machine learning models are no longer restricted to probabilistic ones. Therefore, an advantage of machine learning techniques over statistical ones is that the latter require underlying probabilistic models while the former do not. Even though some machine learning techniques use probabilistic models, the classical statistical techniques are most often too stringent for the oncoming Big Data era, because data sources are increasingly complex and multi-faceted. Prescribing probabilistic models relating variables from disparate data sources that are plausible and amenable to statistical analysis might be extremely difficult if not impossible. Machine learning might be able to provide a broader class of more flexible alternative analysis methods better suited to modern sources of data. It is imperative for statistical agencies to explore the possible use of machine learning techniques to determine whether their future needs might be better met with such techniques than with traditional ones.

There are two main classes of machine learning techniques:

1.Supervised machine learning

2.Unsupervised machine learning.

Examples of supervised learning Logistic regression (statistics) vs Support vector machines (machine learning) . Logistic regression, when used for prediction purposes, is an example of supervised machine learning. In logistic regression, the values of a binary response variable (with values 0 or 1, say) as well as a number of predictor variables (covariates) are observed for a number of observation units. These are called training data in machine learning terminology. The main hypotheses are that the response variable follows a Bernoulli distribution (a class of probabilistic models), and the link between the response and predictor variables is the relation that the logarithm of the posterior odds of the response is a linear function of the predictors. The response variables of the units are assumed to be independent of each other, and the method of maximum likelihood is applied to their joint probability distribution to find the optimal values for the coefficients (these parameterise the aforementioned joint distribution) in this linear function. The particular model with these optimal coefficient values is called the “fitted model,” and can be used to “predict” the value of the response variable for a new unit (or, “classify” the new unit as 0 or 1) for which only the predictor values are known. Support Vector Machines (SVM) are an example of a non-statistical supervised machine learning technique; it has the same goal as the logistic regression classifier just described: Given training data, find the best-fitting SVM model, and then use the fitted SVM model to classify new units. The difference is that the underlying models for SVM are the collection of hyperplanes in the space of the predictor variables. The optimization problem that needs to be solved is finding the hyperplane that best separates, in the predictor space, the units with response value 0 from those with response value 1. The logistic regression optimization problem comes from probability theory whereas that of SVM comes from geometry. Other supervised machine learning techniques mentioned later in this briefing include decision trees, neural networks, and Bayesian networks.

Examples of unsupervised learning Principal component analysis (statistics) vs Cluster analysis (machine learning) . The main example of an unsupervised machine learning technique that comes from classical statistics is principal component analysis, which seeks to “summarize” a set of data points in high-dimensional space by finding orthogonal one-dimensional subspaces along which most of the variation in the data points is captured. The term “unsupervised” simply refers to the fact that there is no longer a response variable in the current setting. Cluster analysis and association analysis are examples of non-statistical unsupervised machine learning techniques. The former seeks to determine inherent grouping structure in given data, whereas the latter seeks to determine co-occurrence patterns of items.

**INTRODUCTION TO PROJECT**

This is the MACHINE LEARNING project to predict the insurance cost according to a persons age(age of primary beneficiary) ,sex(insurance contractor gender), female,&male,bmi(body mass index, Body mass index, providing an understanding of body, weights that are relatively high or low relative to height),smoker(is the person smokes or not),region(the beneficiary's residential area in the US, northeast, southeast, southwest, northwest),charges(charges: Individual medical costs billed by health insurance).The project consist of four models 1.multiple regressor model(MLR) 2.random forest regressor (RFR) 3.multiple linear regressor(MLR) with principal component analysis(PCA) 4.random forest regressor(RFR) with principal component(PCA).

CHAPTER :- 2

MULTIPLE LINEAR REGRESSOR

2.1 Introduction

Regression models are used to describe relationships between variables by fitting a line to the observed data. Regression allows you to estimate how a [dependent variable](https://www.scribbr.com/methodology/independent-and-dependent-variables/) changes as the independent variable(s) change.

Multiple linear regression is used to estimate the relationship between two or more independent variables and one dependent variable. You can use multiple linear regression when you want to know:

1. How strong the relationship is between two or more independent variables and one dependent variable (e.g. how rainfall, temperature, and amount of fertilizer added affect crop growth).
2. The value of the dependent variable at a certain value of the independent variables (e.g. the expected yield of a crop at certain levels of rainfall, temperature, and fertilizer addition).

Assumptions of multiple linear regression

1. Multiple linear regression makes all of the same assumptions as[simple linear regression](https://www.scribbr.com/statistics/simple-linear-regression/):
2. Homogeneity of variance (homoscedasticity): the size of the error in our prediction doesn’t change significantly across the values of the independent variable.
3. Independence of observations: the observations in the dataset were collected using statistically valid methods, and there are no hidden relationships among variables.
4. In multiple linear regression, it is possible that some of the independent variables are actually correlated with one another, so it is important to check these before developing the regression model. If two independent variables are too highly correlated (r2 > ~0.6), then only one of them should be used in the regression model.
5. Normality: The data follows a [normal distribution](https://www.scribbr.com/statistics/normal-distribution/).
6. Linearity: the line of best fit through the data points is a straight line, rather than a curve or some sort of grouping factor.

## To perform a multiple linear regression

### Multiple linear regression formula

The formula for a multiple linear regression is:

Multiple linear regression formula

* **y** = the predicted value of the dependent variable
* **B0**= the y-intercept (value of y when all other parameters are set to 0)
* **B1X1**= the regression coefficient (B1) of the first independent variable (**X1**) (a.k.a. the effect that increasing the value of the independent variable has on the predicted **y** value)
* **…** = do the same for however many independent variables you are testing
* **BnXn** = the regression coefficient of the last independent variable
* **e** = model error (a.k.a. how much variation there is in our estimate of **y**)

To find the best-fit line for each independent variable, multiple linear regression calculates three things:

* The regression coefficients that lead to the smallest overall model error.
* The t-statistic of the overall model.
* The associated [p-value](https://www.scribbr.com/statistics/p-value/) (how likely it is that the t-statistic would have occurred by chance if the null hypothesis of no relationship between the independent and dependent variables was true).

It then calculates the t-statistic and p-value for each regression coefficient in the model.

### Multiple linear regression in R

While it is possible to do multiple linear regression by hand, it is much more commonly done via statistical software. We are going to use R for our examples because it is free, powerful, and widely available.

2.2 Simple linear regression

Simple linear regression is a function that allows an analyst or statistician to make predictions about one variable based on the information that is known about another variable. Linear regression can only be used when one has two continuous variables—an independent variable and a dependent variable. The independent variable is the parameter that is used to calculate the dependent variable or outcome. A multiple regression model extends to several explanatory variables.

Assumptions of simple linear regression

Simple linear regression is a [parametric test](https://www.scribbr.com/statistics/statistical-tests/#parametric), meaning that it makes certain assumptions about the data. These assumptions are:

1. Homogeneity of variance (homoscedasticity): the size of the error in our prediction doesn’t change significantly across the values of the independent variable.
2. Independence of observations: the observations in the dataset were collected using [statistically valid sampling methods](https://www.scribbr.com/methodology/sampling-methods/), and there are no hidden relationships among observations.
3. Normality: The data follows a [normal distribution](https://www.scribbr.com/statistics/normal-distribution/).

Linear regression makes one additional assumption:

1. The relationship between the independent and dependent variable is linear: the line of best fit through the data points is a straight line (rather than a curve or some sort of grouping factor).

2.3 Linearity

Another consideration is decision boundaries. Some algorithms, such as logistic regression or **Support Vector Machine** (**SVM**), can learn linear decision boundaries while others, such as tree-based algorithms, can learn non-linear decision boundaries. While linear decision boundaries are relatively easy to calculate and interpret, you should be aware of errors that linear algorithms will generate in the presence of non-linear relationships.

2.4 Multivariate Normality

The [multivariate normal](https://en.wikipedia.org/wiki/Multivariate_normal_distribution)distribution is a multidimensional generalisation of the one-dimensional [normal distribution](https://en.wikipedia.org/wiki/Normal_distribution). It represents the distribution of a [multivariate random variable](https://en.wikipedia.org/wiki/Multivariate_random_variable)that is made up of multiple random variables that can be correlated with eachother.

Like the normal distribution, the multivariate normal is defined by sets of parameters: the mean vector μμ, which is the expected value of the distribution; and the [covariance](https://en.wikipedia.org/wiki/Covariance)matrix ΣΣ, which measures how dependend two random variables are and how they change together. We denote the covariance between variable XX and YY as C(X,Y)C(X,Y).

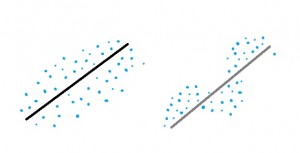
The multivariate normal with dimensionality dd has a [joint probability](https://en.wikipedia.org/wiki/Joint_probability_distribution)density given by:

p(x∣μ,Σ)=1√(2π)d|Σ|exp(−12(x−μ)TΣ−1(x−μ))p(x∣μ,Σ)=1(2π)d|Σ|exp⁡(−12(x−μ)TΣ−1(x−μ))

Where xx a random vector of size dd, μμ is the mean vector, ΣΣ is the ( [symmetric](https://en.wikipedia.org/wiki/Symmetric_matrix), [positive definite](https://en.wikipedia.org/wiki/Positive-definite_matrix)) covariance matrix (of size d×dd×d), and |Σ||Σ| its [determinant](https://en.wikipedia.org/wiki/Determinant). We denote this multivariate normal distribution as:

N(μ,Σ)

2.5 Independence of Error

if your points are following a clear pattern, it might indicate that the errors are influencing each other. The errors are the deviations of an observed value from the true function value. The following image shows two linear regression lines; on the left, the points are scattered randomly. On the right, the points are clearly influencing each other.  
[](https://www.statisticshowto.com/wp-content/uploads/2014/02/Independence-of-Errors.jpg)

2.6 Lack of Multicolinearity

Multicollinearity in regression occurs when predictor variables (independent variables) in the regression model are more highly correlated with other predictor variables than with the dependent variable. Multicollinearity does not adversely affect the regression equation if the purpose of your research is only to predict the dependent variable from a set of predictor variables. In this case the predictions in your regression will still be accurate, and the overall R2 will give you an indication of how well the predictor variables in your model predict the dependent variable. Multicollinearity does not affect the goodness of fit and the goodness of prediction.

     In regression, multicollinearity can be a problem if the purpose of your study is to estimate the contributions of individual predictors. When multicollinearity is present, p values can be misleading and the regression coefficients. confidence intervals will be very wide and may vary dramatically with the addition or exclusion of just one case/participant. If this is the case, removing any highly correlated terms from the model will greatly affect the estimated coefficients of the other highly correlated terms. Multicollinearity inflates the variances of the parameter estimates. This may lead to lack of statistical significance of individual independent variables even though the overall model may be significant. This is especially true for small and moderate sample sizes. Such problems will result in incorrect conclusions about relationships between independent and dependent variables. This is a mistake you don.t want to make in your study. Contact us today and we will evaluate your regression analysis for multicollinearity.

2.7 Dummy Variable

A dataset may contain various type of values, sometimes it consists of categorical values. So, in-order to use those categorical value for programming efficiently we create dummy variables. A dummy variable is a binary variable that indicates whether a separate categorical variable takes on a specific value.

Step-by-step Approach:

* Import necessary modules
* Consider the data
* Perform operations on data to get dummies

2.8 Backward Elimination

Backward elimination is a feature selection technique while building a machine learning model. It is used to remove those features that do not have a significant effect on the dependent variable or prediction of output. There are various ways to build a model in Machine Learning, which are:

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

Above are the possible methods for building the model in Machine learning, but we will only use here the Backward Elimination process as it is the fastest method.

2.9 Importing the libraries

Following are the necessary libraries used to import for building MLR model

PYTHON LIBRARIES---------

1.import pandas as pd

2.import numpy as np

3.import matplotlib.pyplot as plt

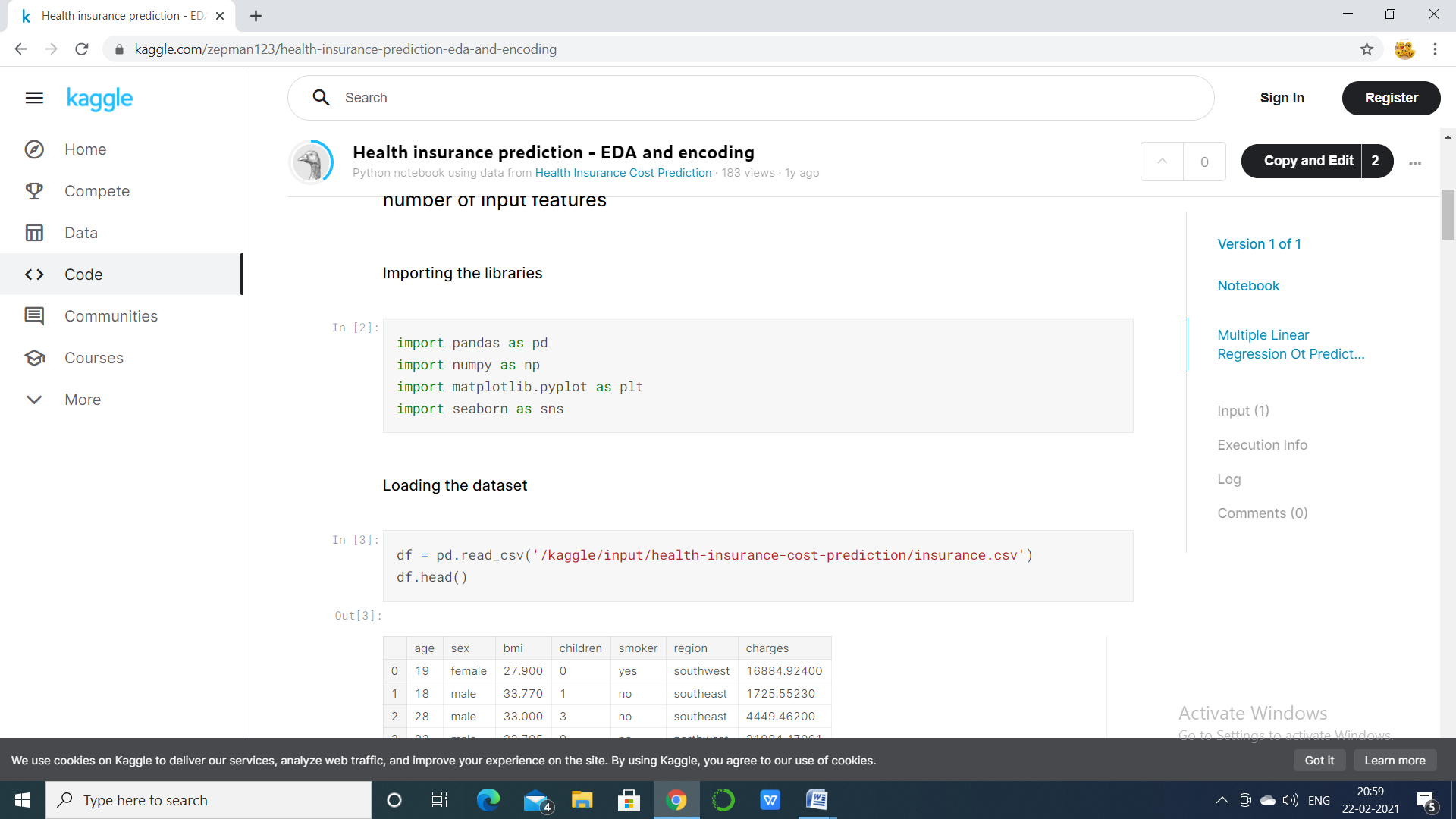
4.from sklearn.compose import ColumnTransformer

5.from sklearn.preprocessing import OneHotEncoder

6.from sklearn.metrics import mean\_absolute\_error,mean\_squared\_error,r2\_score

7.from sklearn.model\_selection import train\_test\_split

8.from sklearn.linear\_model import LinearRegression



2.10 Encoding categorical data

Label Encoding

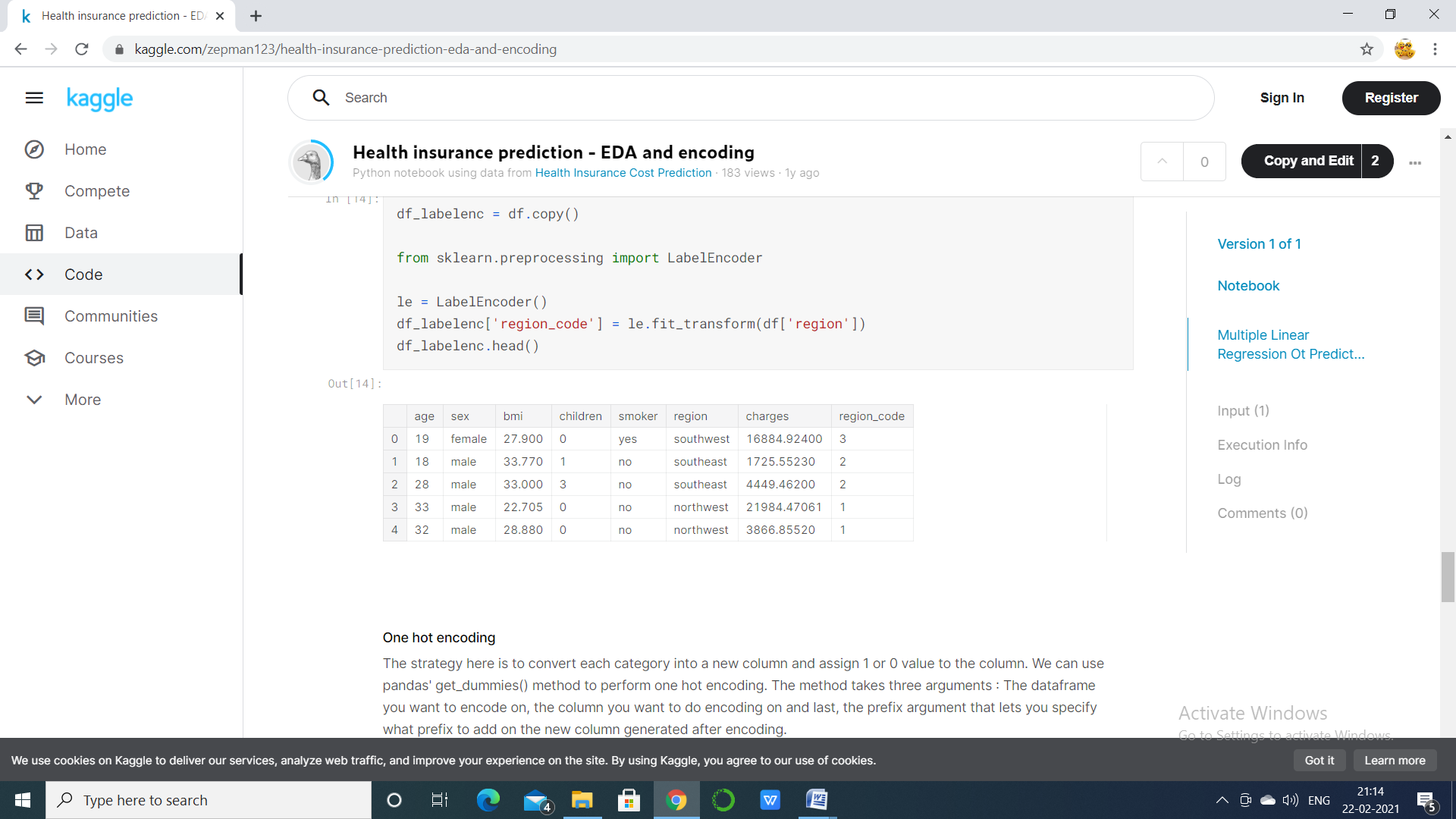
Label encoding is pretty much intuitive and straight-forward and may give you a good performance from your learning algorithm, but it has as disadvantage that the numerical values can be misinterpreted by the algorithm. Should the carrier US (encoded to 8) be given 8x more weight than the carrier AS (encoded to 1) ?

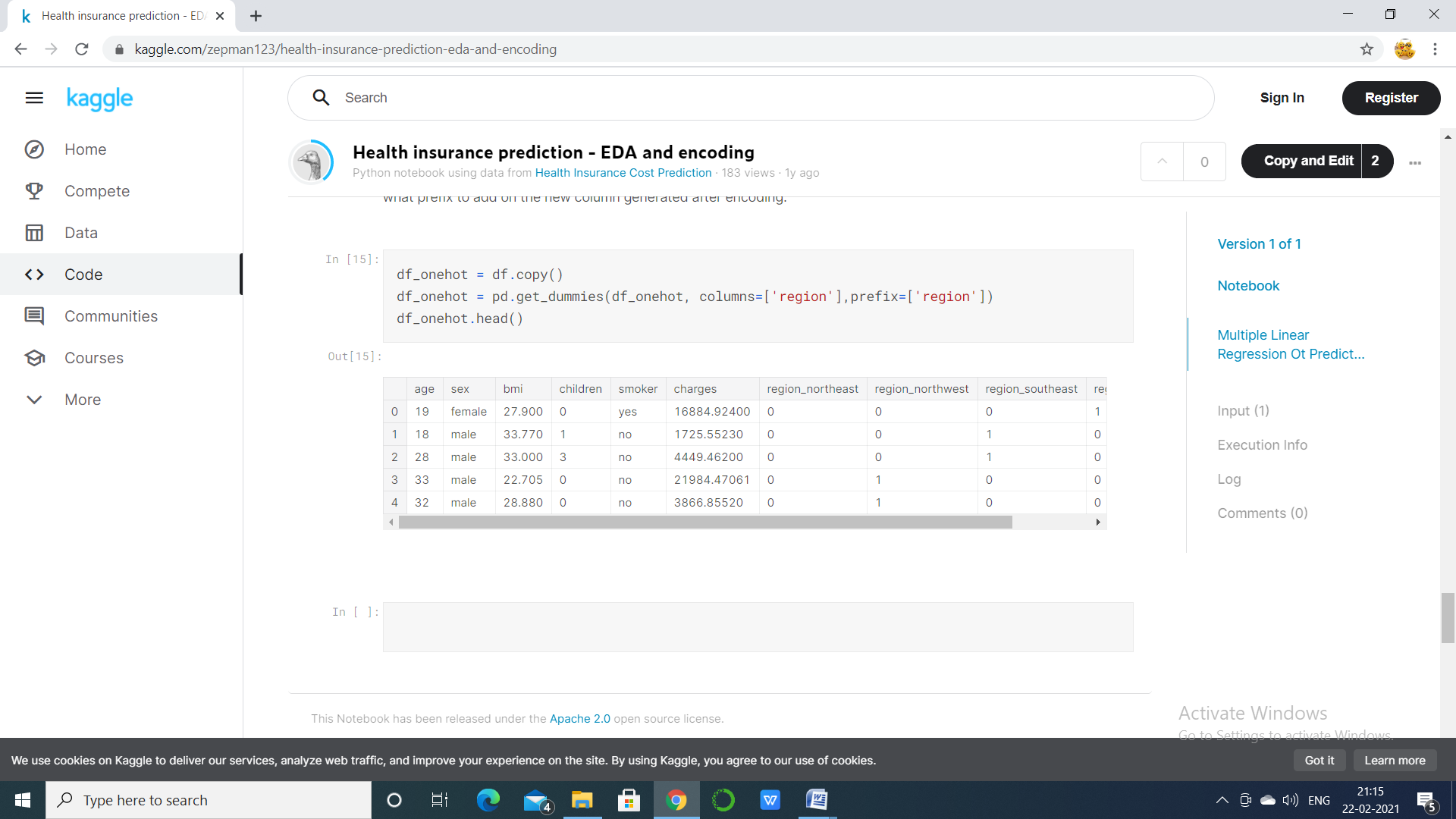
To solve this issue there is another popular way to encode the categories via something called one-hot encoding.

### One-Hot encoding

The basic strategy is to convert each category value into a new column and assign a 1 or 0 (True/False) value to the column. This has the benefit of not weighting a value improperly.

There are many libraries out there that support one-hot encoding but the simplest one is using pandas' .get\_dummies() method.

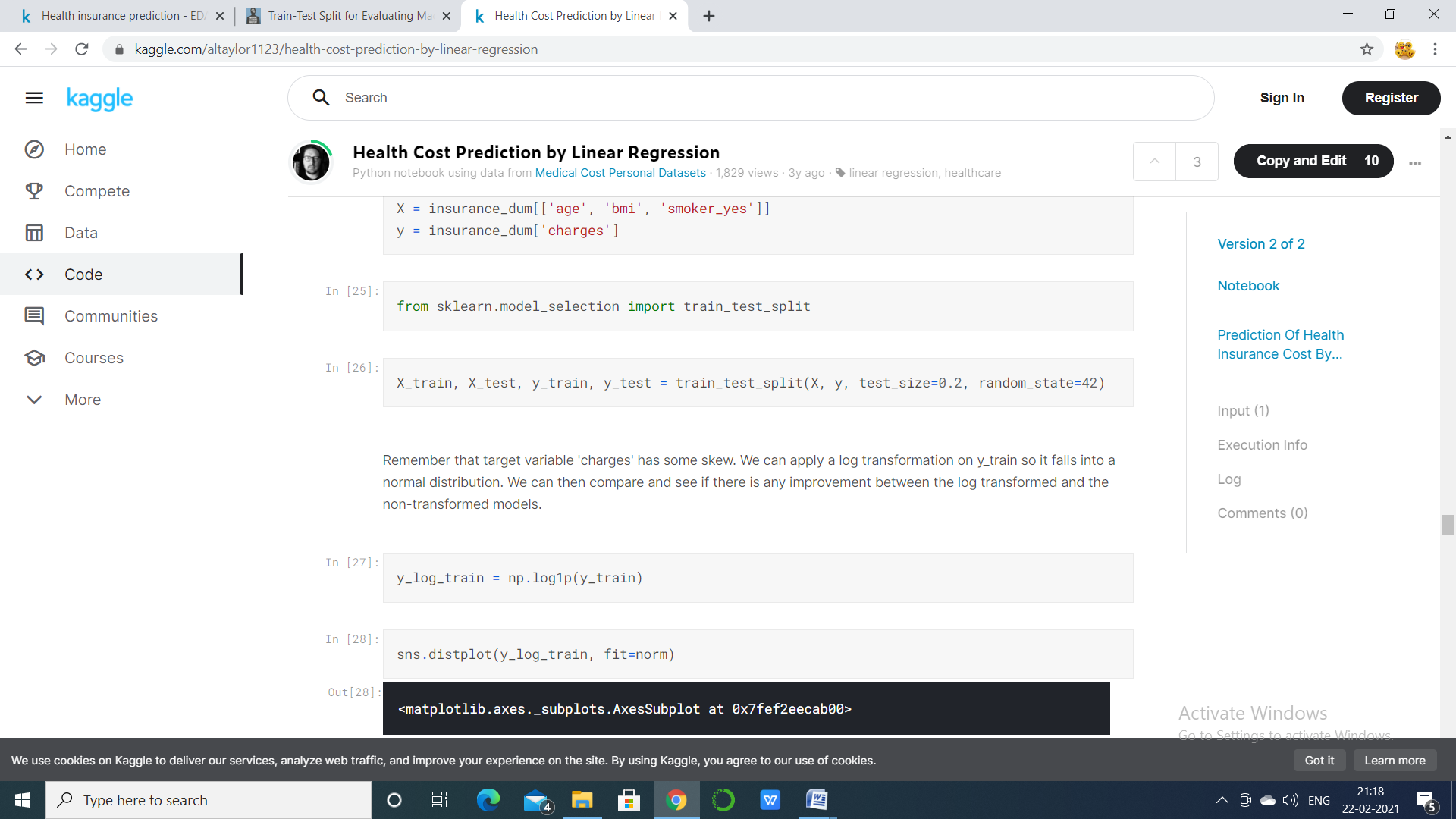
This function is named this way because it creates dummy/indicator variables (1 or 0). There are mainly three arguments important here, the first one is the DataFrame you want to encode on, second being the columns argument which lets you specify the columns you want to do encoding on, and third, the prefix argument which lets you specify the prefix for the new columns that will be created after encoding. 



2.11 Splitting the dataset into the training set and test set

The train-test split procedure is used to estimate the performance of machine learning algorithms when they are used to make predictions on data not used to train the model.

It is a fast and easy procedure to perform, the results of which allow you to compare the performance of machine learning algorithms for your predictive modeling problem. Although simple to use and interpret, there are times when the procedure should not be used, such as when you have a small dataset and situations where additional configuration is required, such as when it is used for classification and the dataset is not balanced.



2.12 Predicting the test set result

1.mae=mean\_absolute\_error(y\_te,y\_pred)=3994.4787473981983

2.mse=mean\_squared\_error(y\_te,y\_pred)=32088433.293003373

3.rmse=np.sqrt(mse)=5664.665329302638

4.r2=r2\_score(y\_te,y\_pred)=0.7983507767817666

The [coefficient of determination](https://www.investopedia.com/terms/c/coefficient-of-determination.asp) (R-squared) is a statistical metric that is used to measure how much of the variation in outcome can be explained by the variation in the independent variables. R2 always increases as more predictors are added to the MLR model, even though the predictors may not be related to the outcome variable.

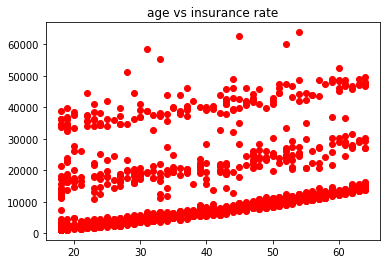
R2 by itself can't thus be used to identify which predictors should be included in a model and which should be excluded. R2 can only be between 0 and 1, where 0 indicates that the outcome cannot be predicted by any of the independent variables and 1 indicates that the outcome can be predicted without error from the independent variables.1

﻿

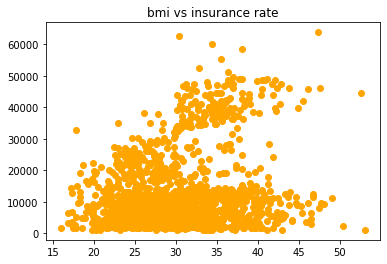
When interpreting the results of multiple regression, beta coefficients are valid while holding all other variables constant ("all else equal"). The output from a multiple regression can be displayed horizontally as an equation, or vertically in the table form.

2.13 Comparing the test set with predicted value

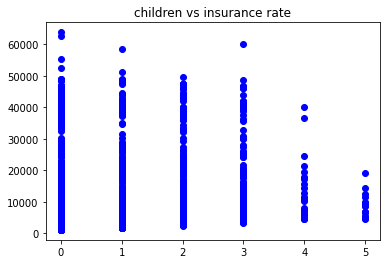
1.age vs insurance---



2.BMI vs Insurance rate..



3. Children vs insurance



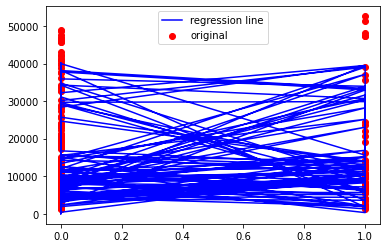
4. Regression line vs original line

plt.scatter(x\_te[:,3],y\_te,c='red',label='original')

plt.plot(x\_te[:,3],y\_pred,c='blue',label='regression line')

plt.legend()

plt.show()



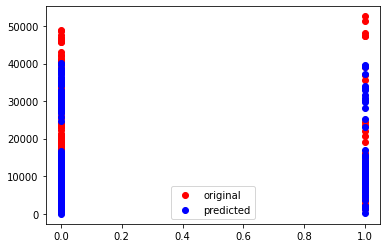
5.Original vs predicted

plt.scatter(x\_te[:,3],y\_te,c='red',label='original')

plt.scatter(x\_te[:,3],y\_pred,c='blue',label='predicted')

plt.legend()

plt.show()



CHAPTER:- 3

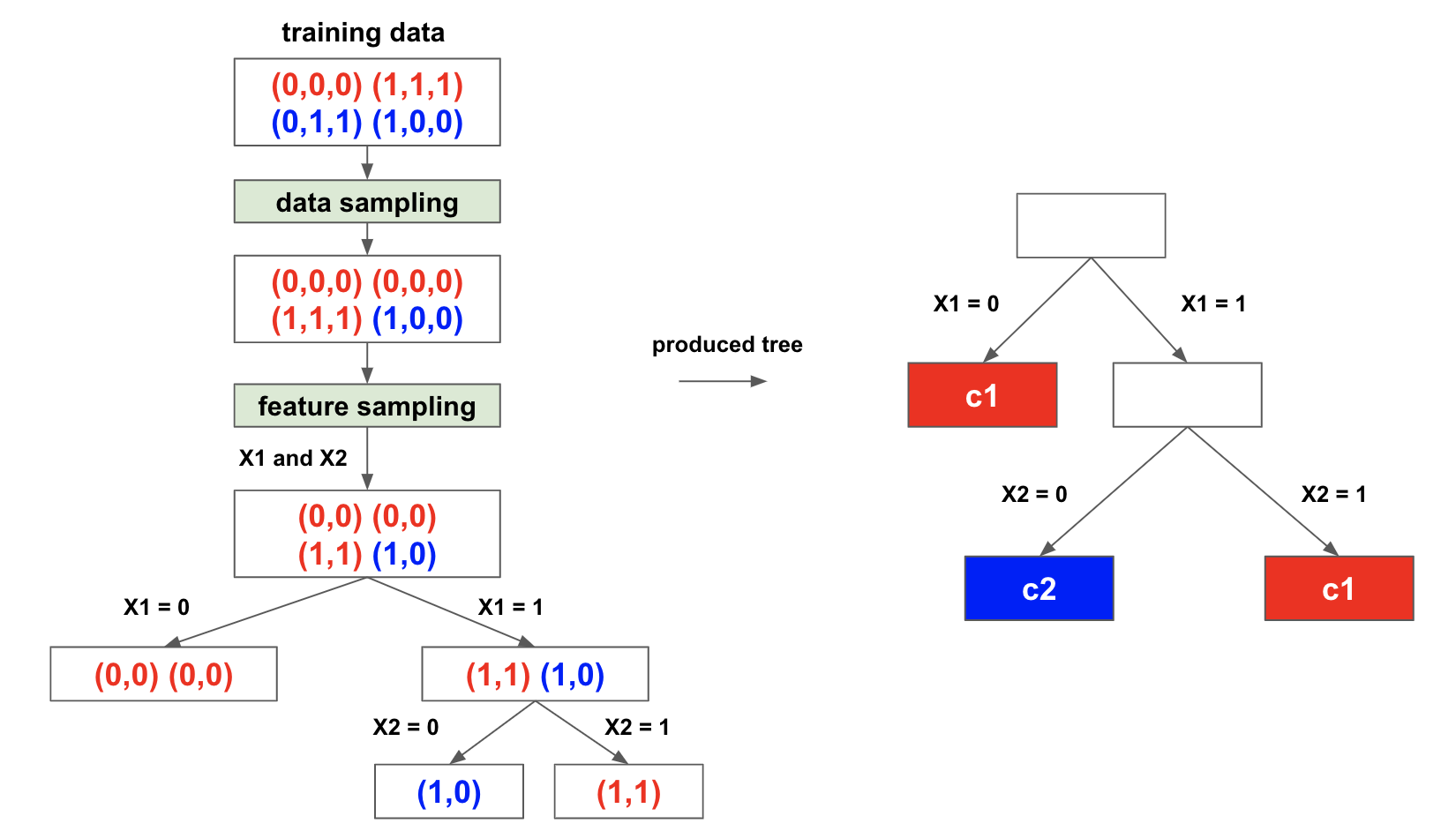
RANDOM FOREST REGRESSOR

3.1 Introduction

Random forests are popularly applied to both data science competitions and practical problems. They are often accurate, do not require feature scaling, categorical feature encoding, and need little parameter tunning. They can also be more interpretable than other complex models such as neural networks.

A random forest consists of multiple random decision trees. Two types of randomnesses are built into the trees. First, each tree is built on a random sample from the original data. Second, at each tree node, a subset of features are randomly selected to generate the best split.

The figure demonstrates to build a random forest tree.



3.2 Ensemble learning

The main motto of ensemble methods is as follows:

1. To decrease the variance(bagging).
2. To decrease the bias(boosting)
3. To improve predictions(stacking)

The Ensemble method can be applied in two ways:

1. Sequential: In this method, the dependency of base learners are exploited. This involves the evaluation and reweighing of the unimportant examples. e.g. Adaboost.
2. Parallel: In this method, the independence of the base learners are exploited. This involves the evaluation by simply averaging the outputs of the base learner. e.g. Random forest.

The ensemble can be of two types :

1. Homogeneous ensemble: In this, we have single base learning algorithm like random forest which only uses the decision tree algorithm.
2. Heterogeneous ensemble : In this, we have different base estimators algorithms.

Techniques used in the ensemble learning:

1. Bagging : Bagging stands for bootstrap aggregation. One way to reduce the variance of an estimate is to average together multiple estimates. For aggregating the outputs of base learners, bagging uses ‘voting for classification’ and ‘averaging for regression’.
2. Boosting: Boosting refers to a family of algorithms that are able to convert weak learners to strong learners. The predictions are then combined through a weighted majority vote (classification) or a weighted sum (regression) to produce the final prediction.
3. Stacking: Stacking is an ensemble learning technique that combines multiple classifications or regression models via a meta-classifier or a meta-regression*.* The base level models are trained based on a complete training set, then the meta-model is trained on the outputs of the base level model as features.

3.3 Decision tree Regressor

Decision Tree is a decision-making tool that uses a flowchart-like tree structure or is a model of decisions and all of their possible results, including outcomes, input costs and utility.

Decision-tree algorithm falls under the category of supervised learning algorithms. It works for both continuous as well as categorical output variables.

The branches/edges represent the result of the node and the nodes have either:

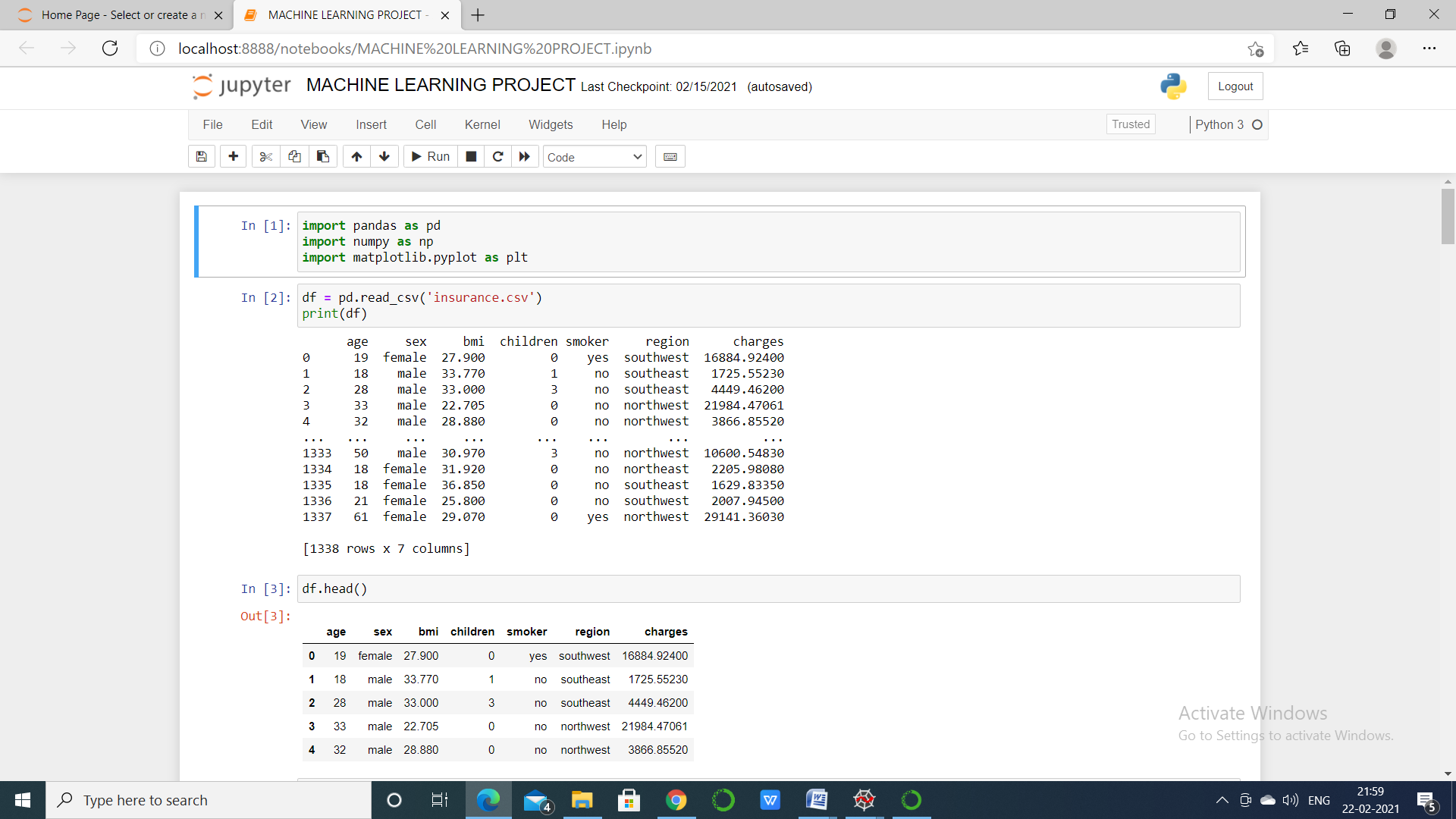
1. Conditions [Decision Nodes]
2. Result [End Nodes]  
   Decision tree regression observes features of an object and trains a model in the structure of a tree to predict data in the future to produce meaningful continuous output. Continuous output means that the output/result is not discrete, i.e., it is not represented just by a discrete, known set of numbers or values.

3.4 Importing the libraries

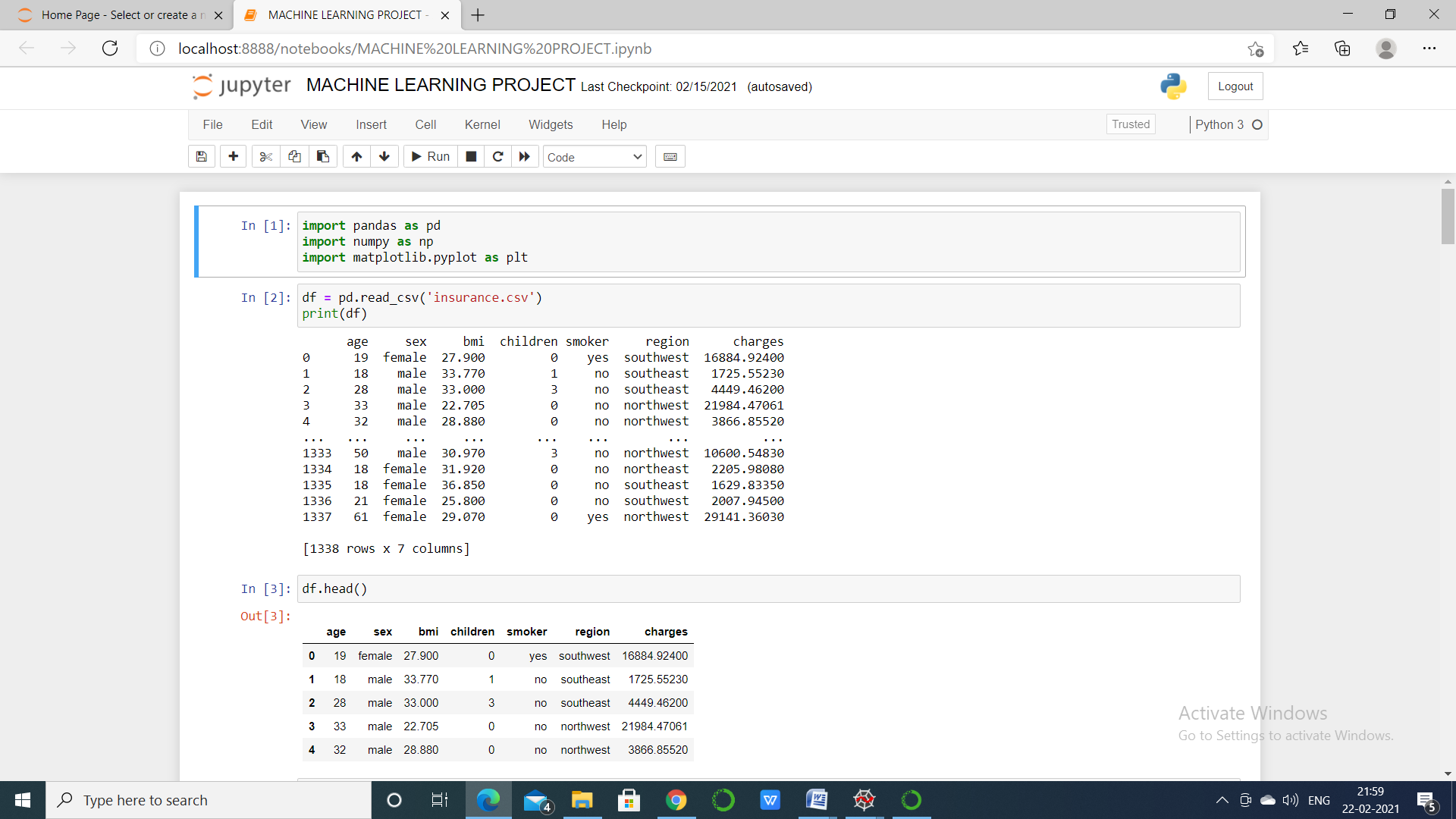
import pandas as pd

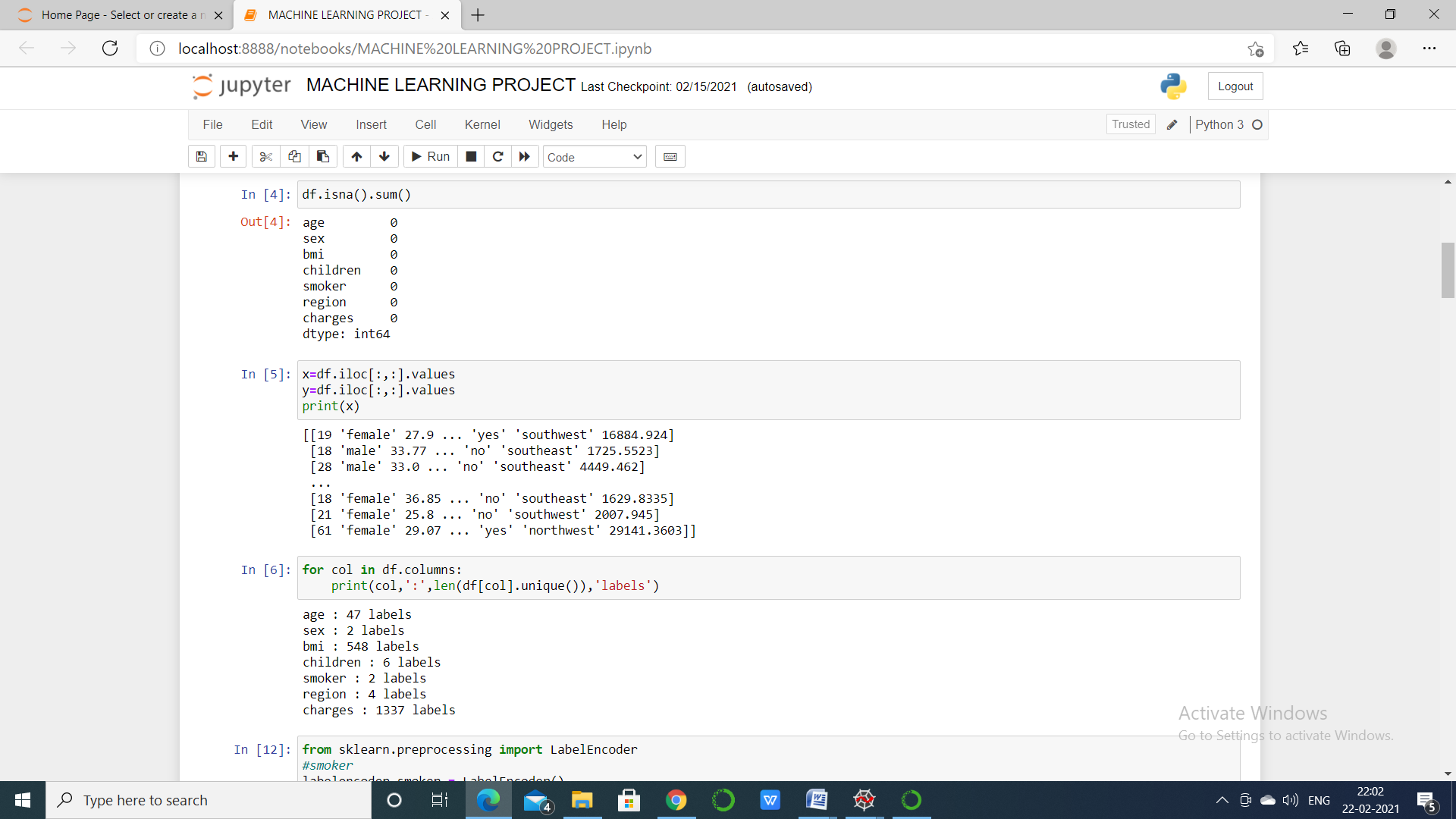
import numpy as np

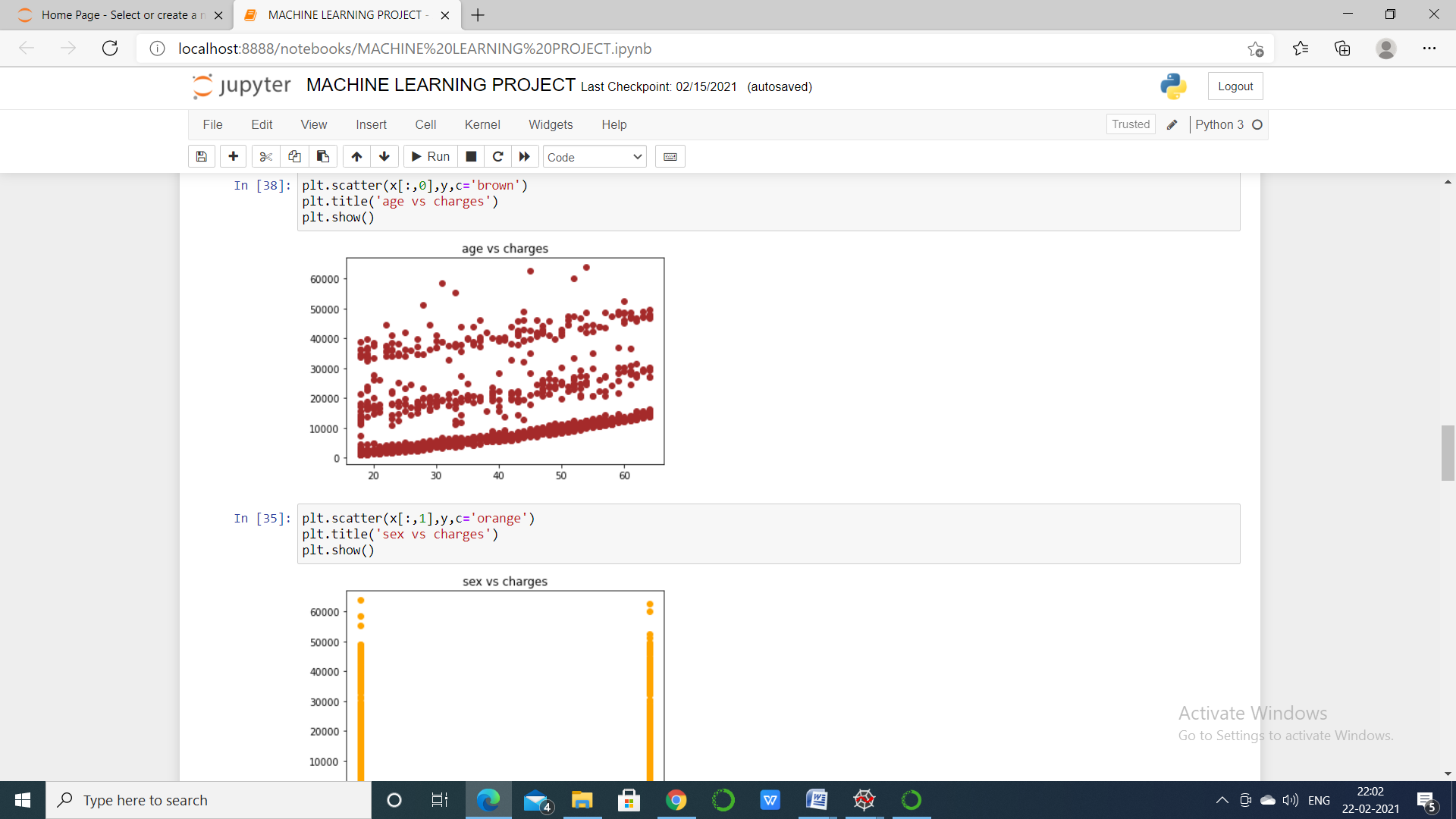
import matplotlib.pyplot as plt

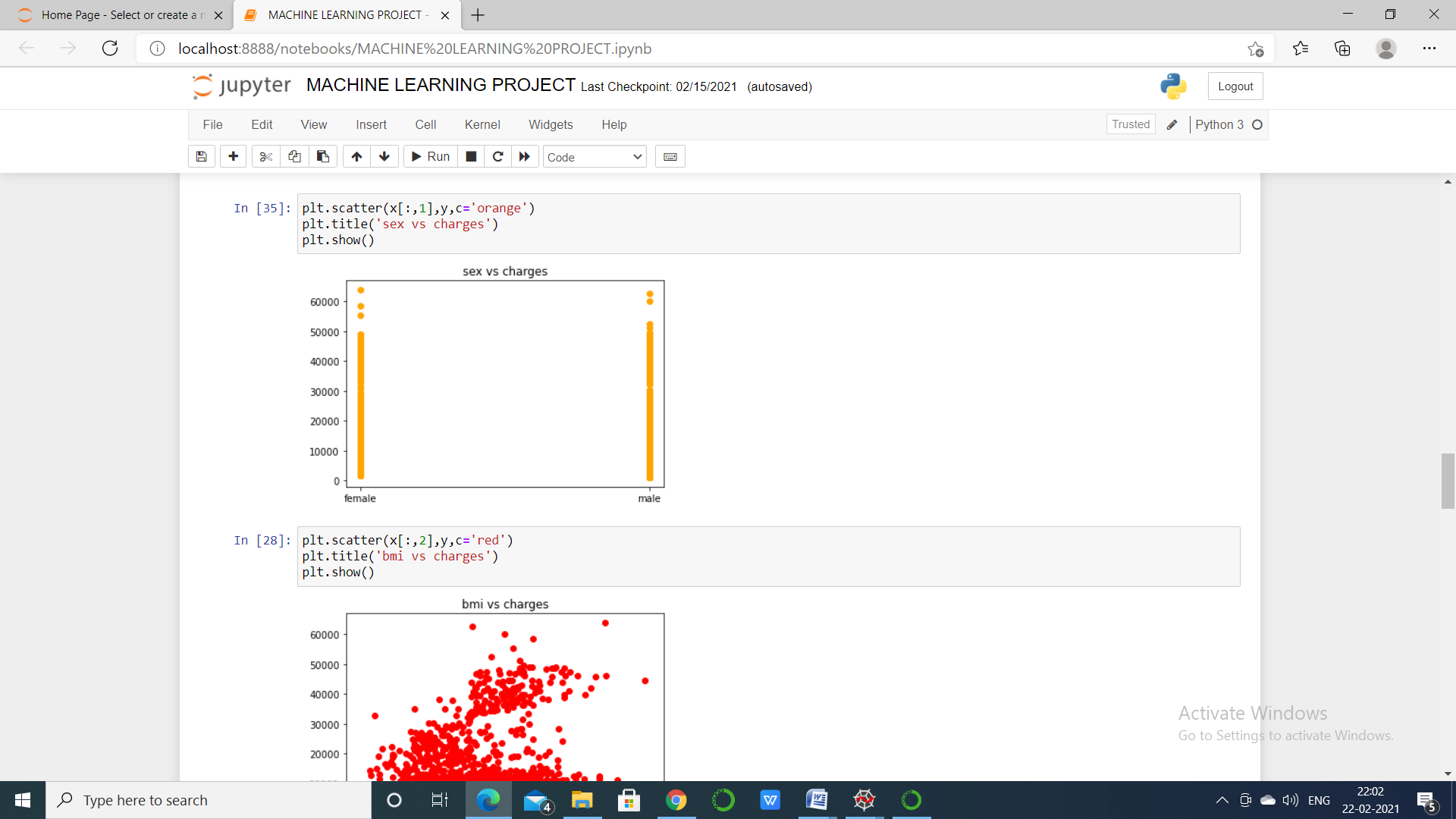


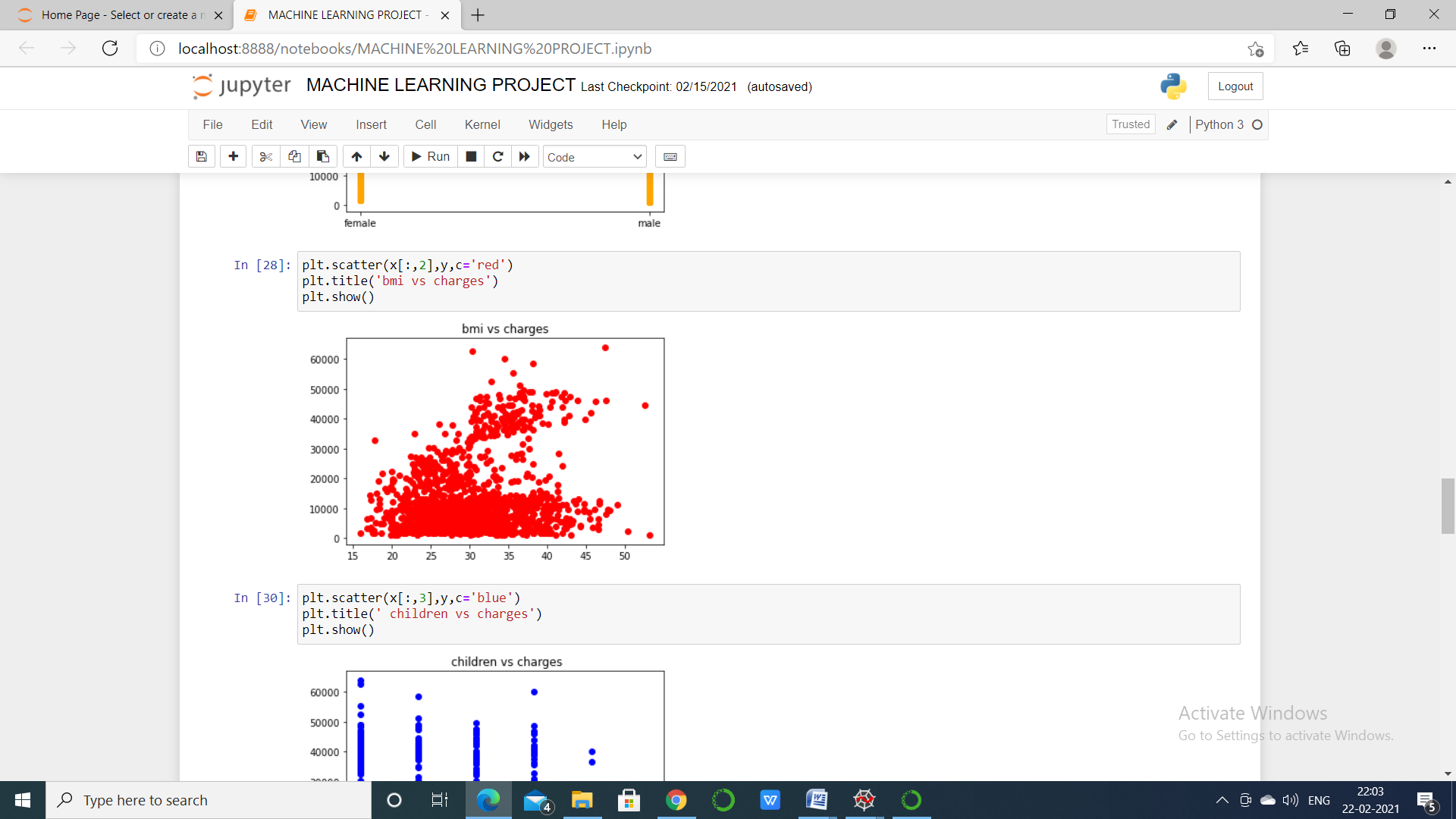
3.5 Importing the dataset

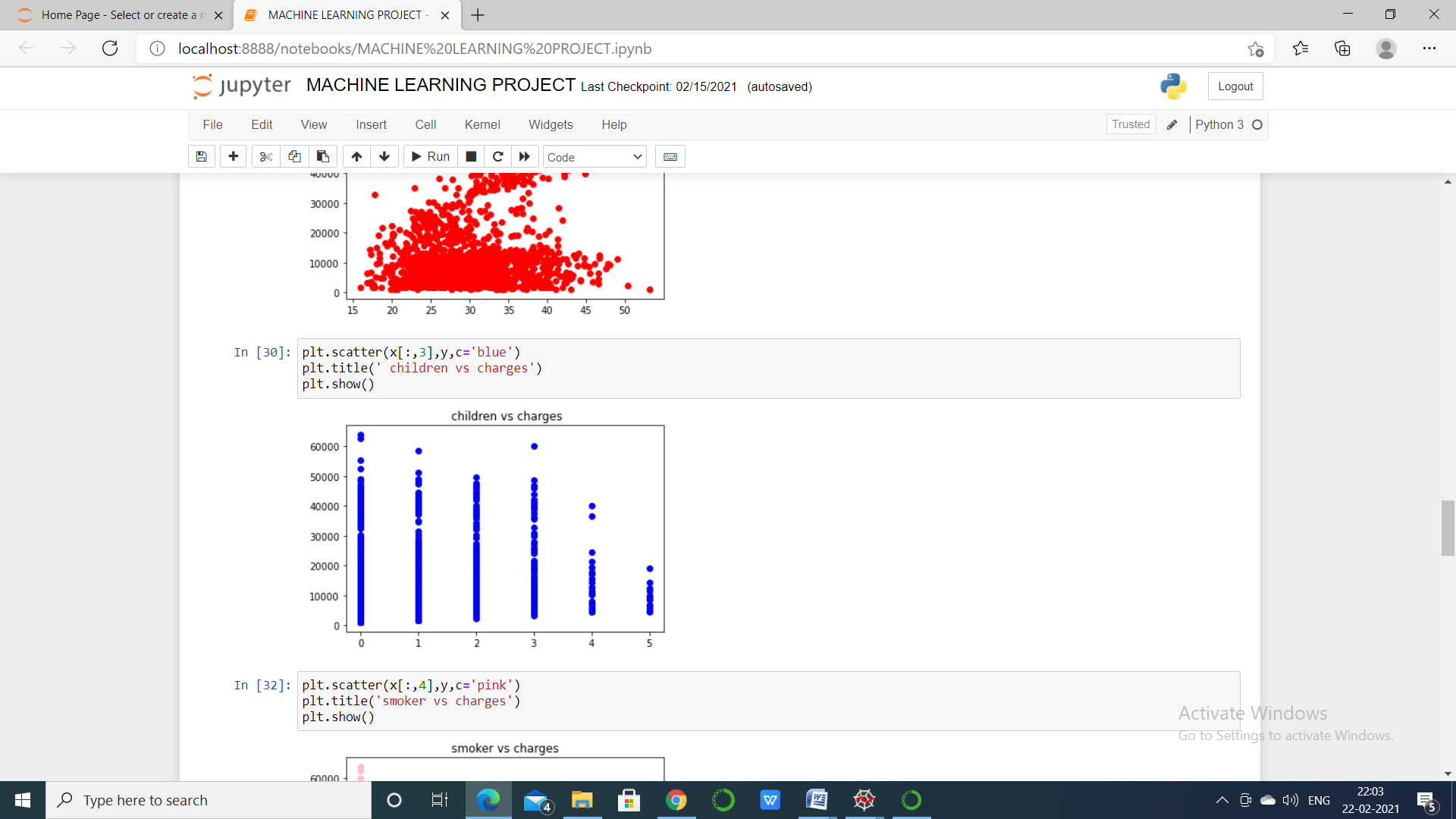


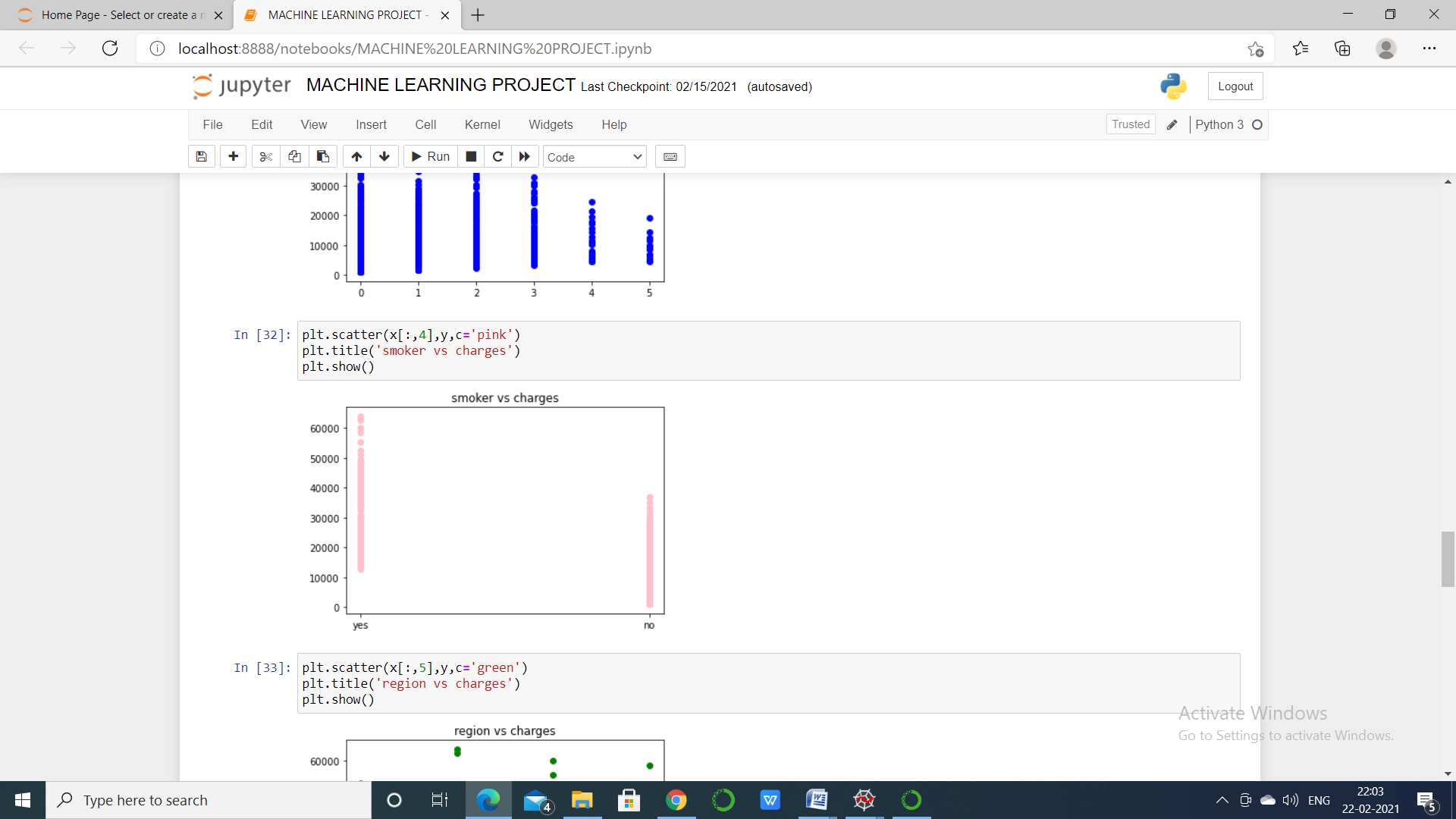


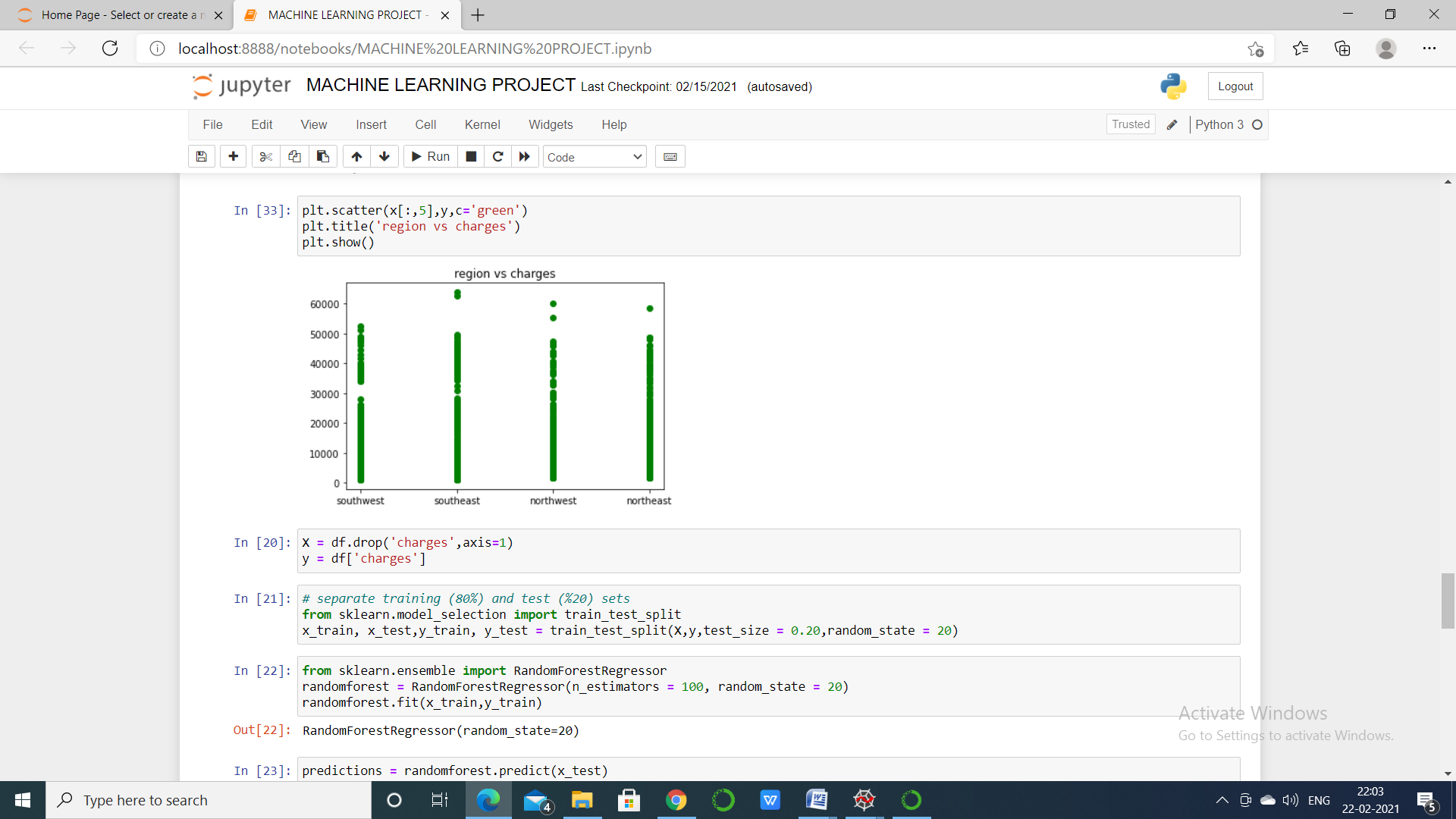








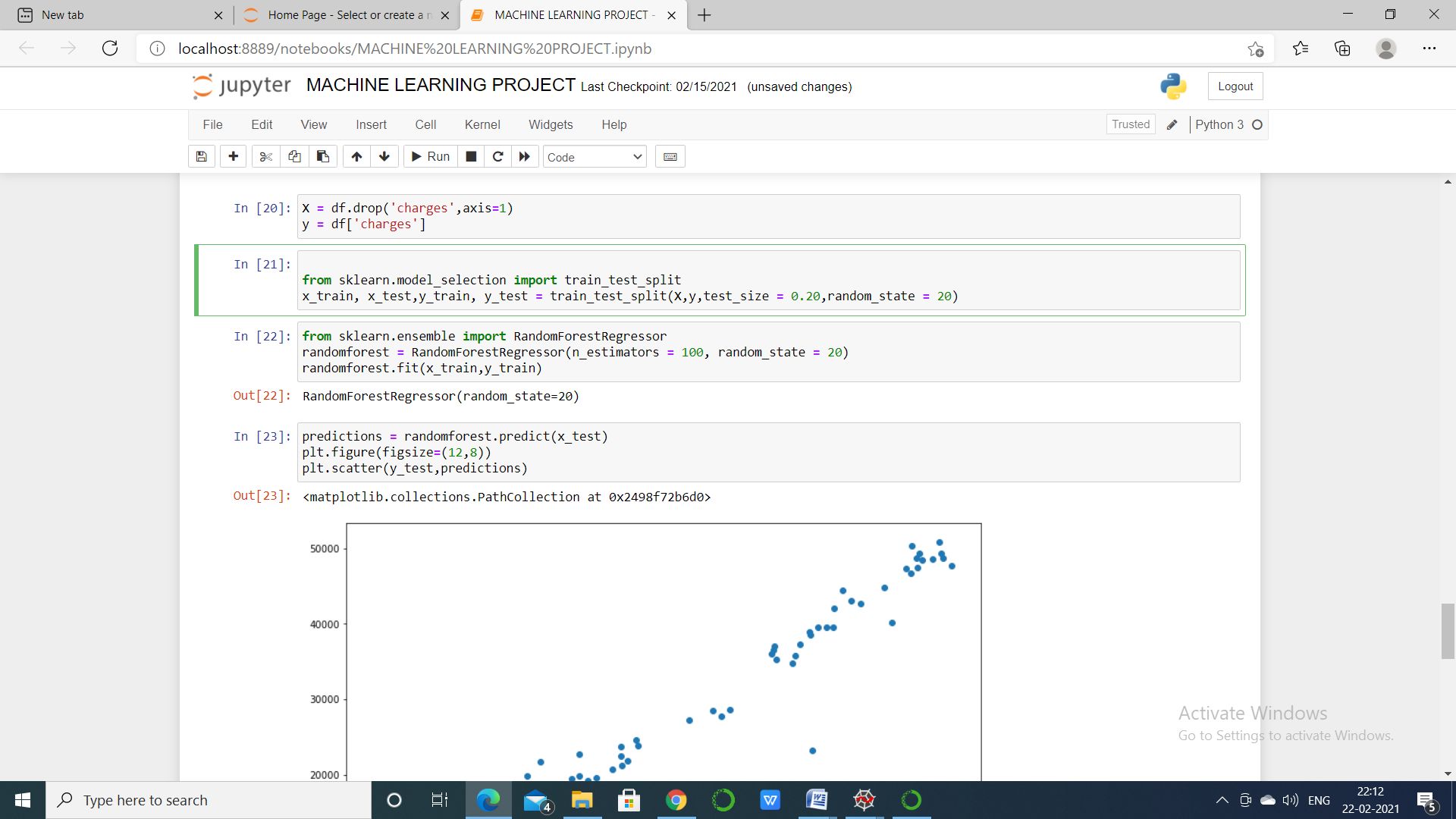




3.6 Splitting the data set into training set and test set

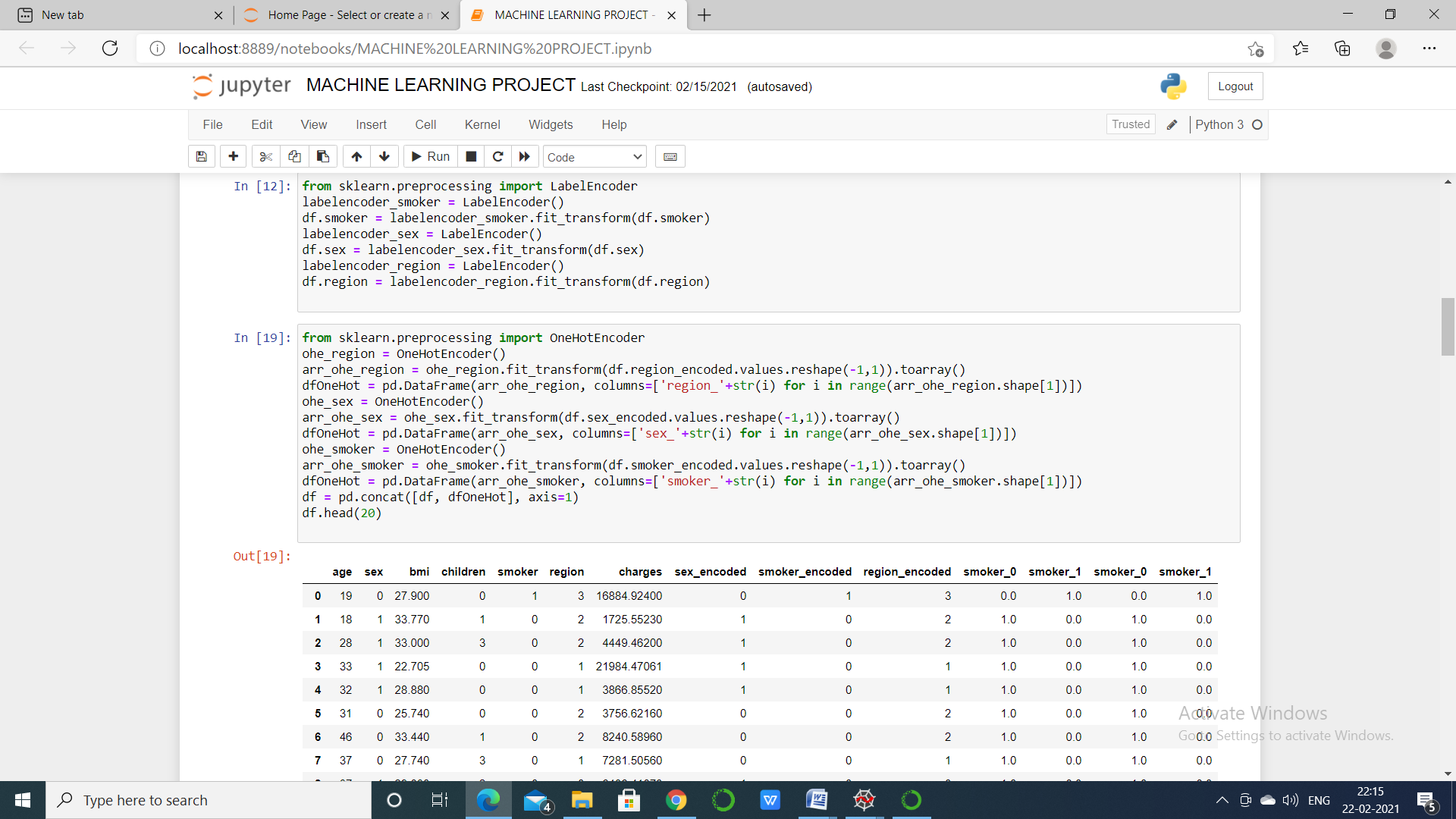
Separating data into training and testing sets is an important part of evaluating data mining models. ... By using similar data for training and testing, you can minimize the effects of data discrepancies and better understand the characteristics of the model.

Use train\_test\_split() to get training and test sets. Control the size of the subsets with the parameters train\_size and test\_size. Determine the randomness of your splits with the random\_state parameter. Obtain stratified splits with the stratify parameter.

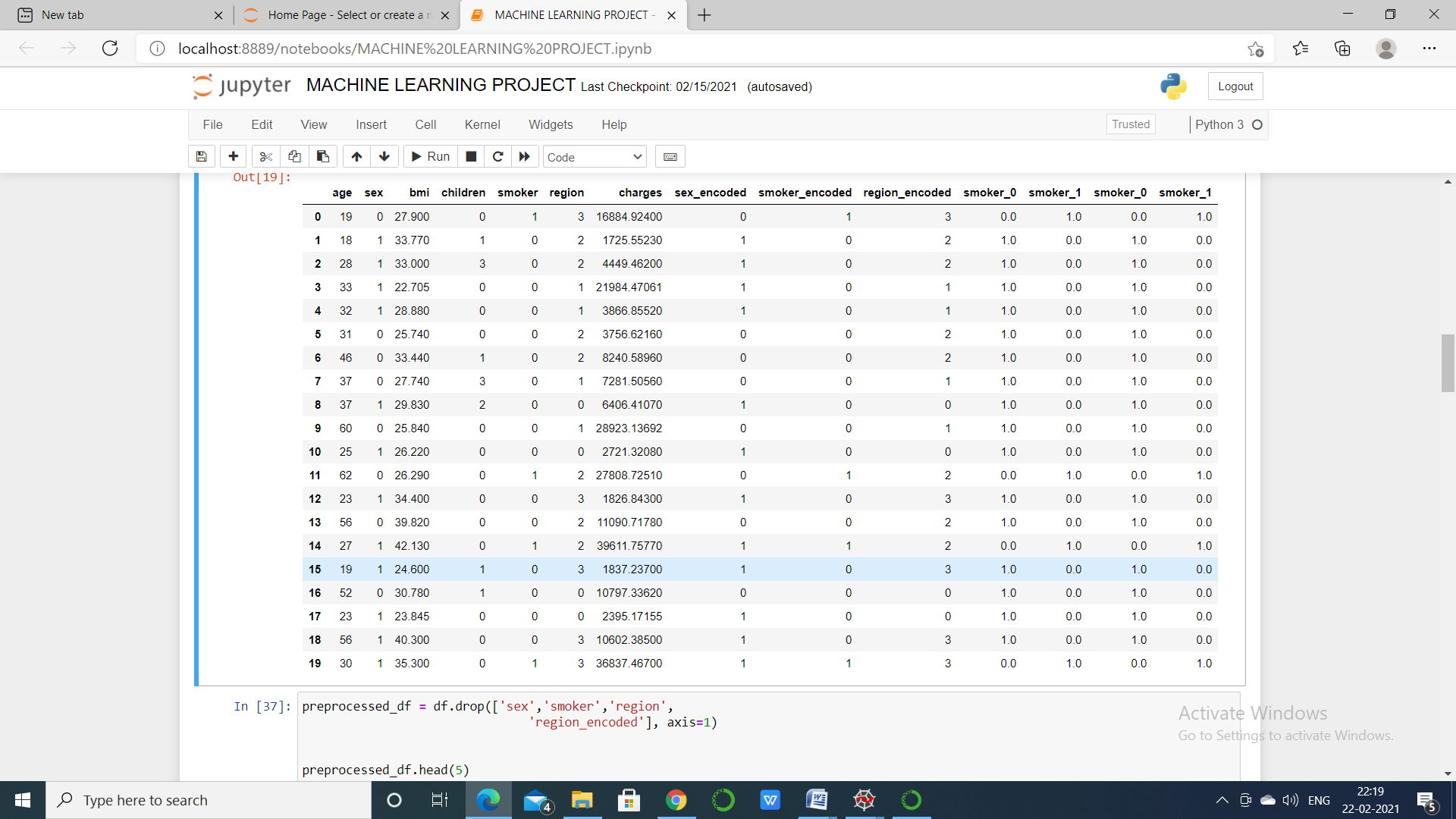


3.7 Encoding the categorical value

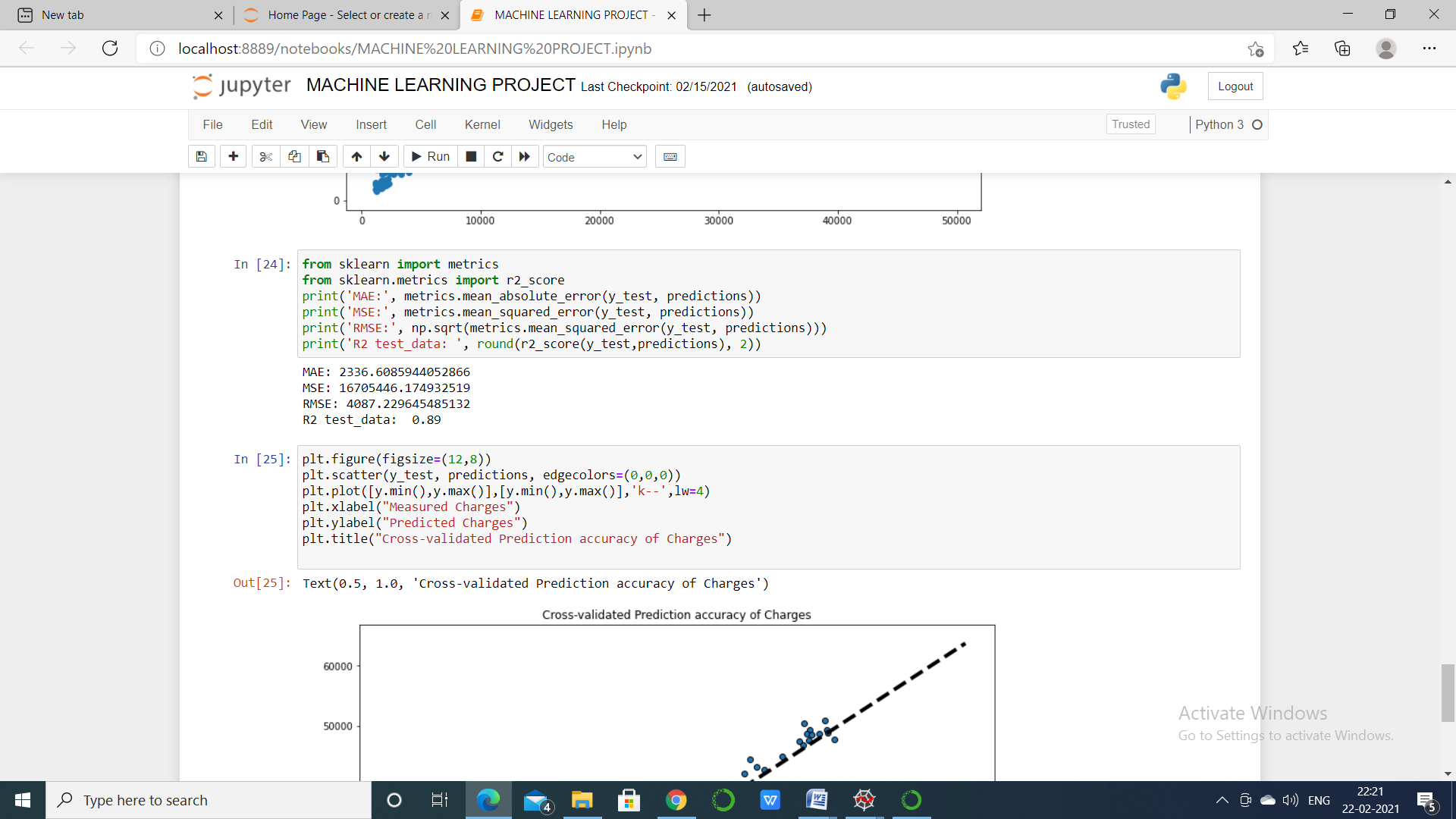
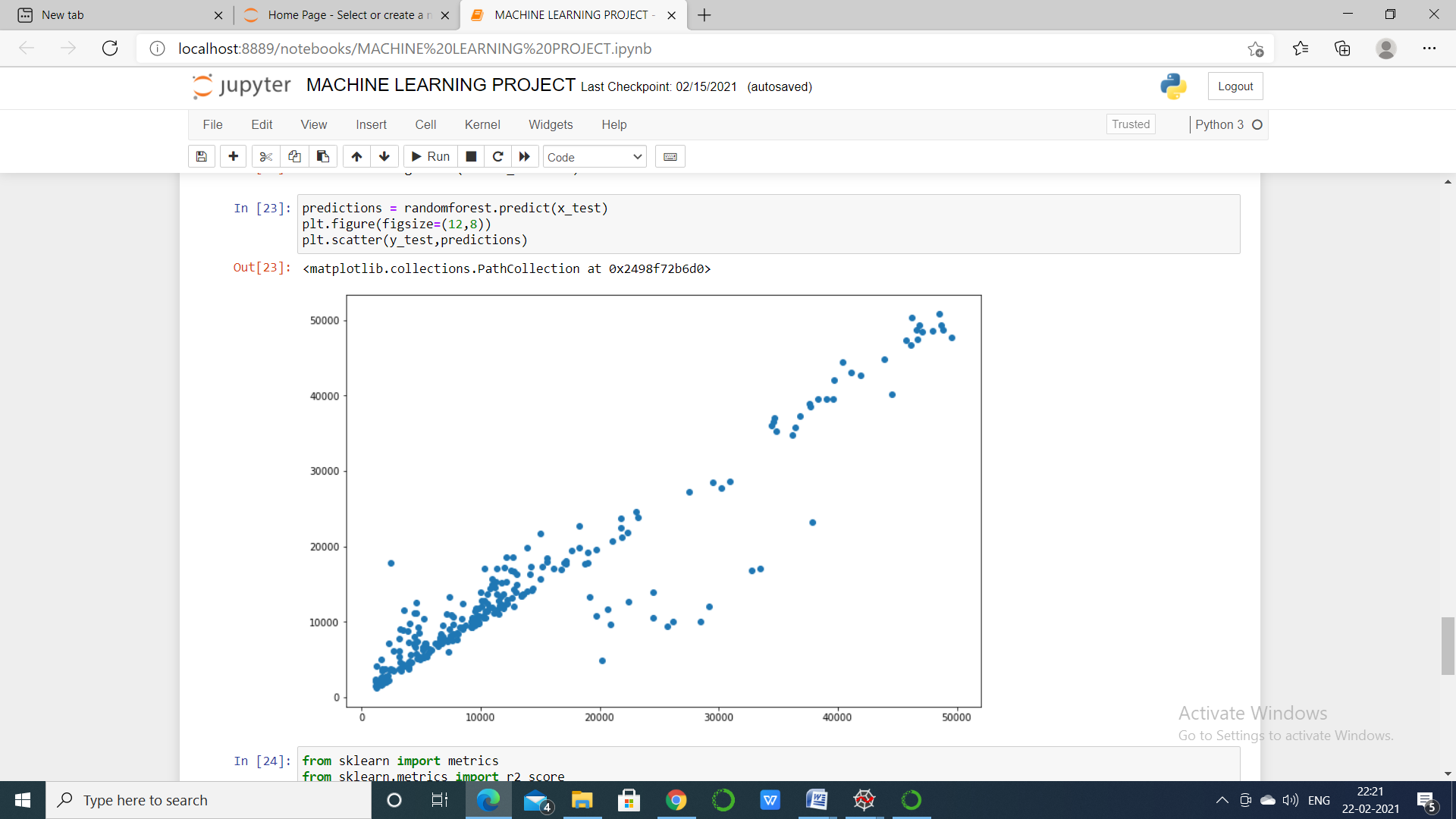
This help to convert the categorical values into numeric(dummy variables)



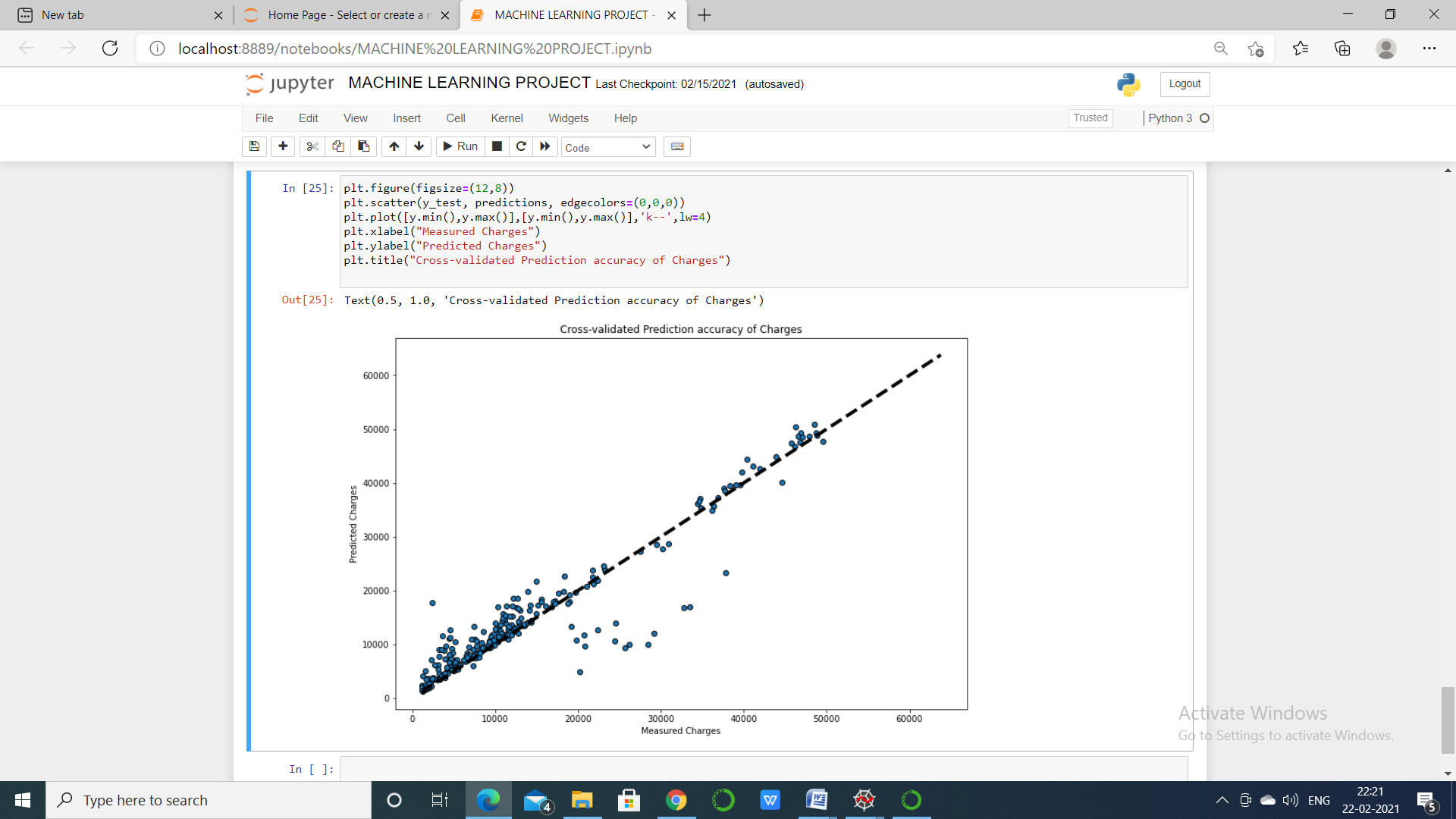
Dummy variables



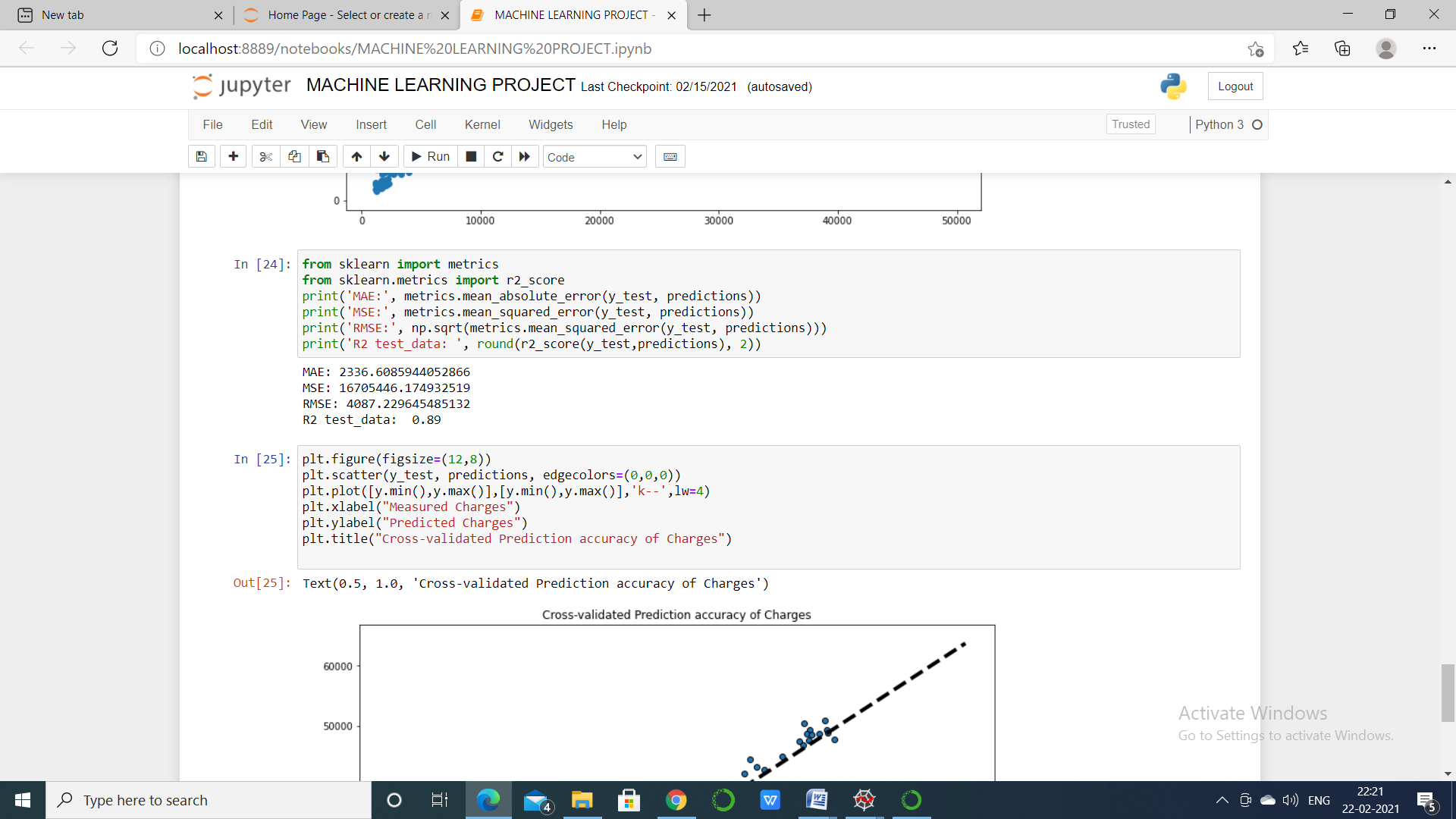
3.8 Predicting the result



3.9 Comparing the real values with predicted values



3.10 Visualizing the random forest regression result



CHAPTER :- 4

PRINCIPAL COMPONENT ANALYSIS

4.1 Introduction

**Principal Component Analysis (PCA)**is a statistical procedure that uses an orthogonal transformation which converts a set of correlated variables to a set of uncorrelated variables. PCA is a most widely used tool in exploratory data analysis and in machine learning for predictive models. Moreover, PCA is an unsupervised statistical technique used to examine the interrelations among a set of variables. It is also known as a general factor analysis where regression determines a line of best fit.

GOALS..

Identify patters in data.Detect the correlation between variables.Reduce the dimensions of a d-dimensional dataset by projecting it onto a k-dimensional subspace(where k<d)

4.2 PCA linear regressor

The goal of PCA is to decompose a matrix into a linear combination of variables that contain most of the information in the matrix. Suppose for sake of argument that we're doing PCA on an input matrix rather than its covariance matrix, and the columns X1,X2,...,XnX1,X2,...,Xn of the matrix are variables of interest. Then intuitively it seems that the PCA procedure is similar to a linear regression where one uses a linear combination of the variables to predict the entries in the matrix. Imagine enumerating the (infinite) space of all linear combinations of the variables X1,X2,...,XnX1,X2,...,Xn of a matrix of data and doing linear regression on each such combination to measure how much of the rows of the matrix the combination can 'explain'. Is there an interpretation of what PCA doing in terms of this operation? I.e. how in this procedure PCA would select the 'best' linear combinations? I realize this procedure is obviously not computationally feasible, I only present it to try to make the link between PCA and linear regression. This procedure works directly with linear combinations of columns of a matrix so it does not require them to be orthogonal

4.3 PCA with missing value

Principal Component Analysis (PCA) is a method of statistical analysis which can be used to attempt to ‘explain’ patterns in the data and estimate the level of structure in data. Input to the PCA can be any set of numerical variables, however they should be scaled to each other and traditional PCA will not accept any missing data points.

Data points will be scored by how well they fit into a principal component (PC) based upon a measure of variance within the dataset. Each data point will have a PC that it best fits. In this way PCA can be seen as a kind of clustering analysis.

Each PC will explain a proportion of the structure of the data (if there is any!). The components that explain 85% of the variance (or where the explanatory data is found) can be assumed to be the most important data points.

As mentioned above, traditional PCA does not accept missing data points, however a package in R called [pcaMethods](https://href.li/?https://www.google.com/url?sa=t&rct=j&q=&esrc=s&source=web&cd=2&cad=rja&uact=8&ved=0CCYQFjABahUKEwjZ_LCiiJfHAhWlCtsKHWNCAmU&url=http%3A%2F%2Fwww.bioconductor.org%2Fpackages%2Frelease%2Fbioc%2Fvignettes%2FpcaMethods%2Finst%2Fdoc%2FpcaMethods.pdf&ei=5rDEVdnxKqWV7AbjhImoBg&usg=AFQjCNHrwt2n7Seef5C49YqmrTfnp-aFsA&sig2=Y9lAXk2rP1UJqQgzxM_HPQ&bvm=bv.99804247,d.ZGU) implements a number of optional estimation methods.Missing scores are estimated by projecting known scores back into the ‘principal space’ i.e. if there is a certain amount of variation exhibited in x, y is likely to vary this way based of previous data. Principal subspace is the area around the model i.e. if it falls within the significance area of a linear regression.

4.4 Format for (MLR with PCA)

Standardize the data.Obtain the Eigenvectors and Eigenvalues from the covariance matrix or correlation matrix, or perform Singular Vector Decomposition.Sort eigenvalues in descending order and choose the k eigenvectors that corresponds to the k largest eigenvalues where k is the number of dimensions of the new feature subspace(k<d)Construct the projection matrix W from the selected eigenvectors.Transform the original dataset X via W to obtain a k-dimensional feature subspaceY

4.5 Importing libraries

import pandas as pd

import numpy as np

import matplotlib.pyplot as plt

import seaborn as sns

%matplotlib inline

4.6 Complete algorithm of MLR with PCA

### Step 1: Get your data

### Step 2: Give your data a structure

### Step 3: Standardize your data

### Step 4: Get Covariance of Z

### Step 5: Calculate Eigen Vectors and Eigen Values

.Step 6: Sort the Eigen Vectors

Step 7: Calculate the new features

### Step 8: Drop unimportant features from the new set

CHAPTER:- 5

CONCLUSION

In this article, we have explored the basics of the Multiple linear regression model ,Random Forest Regressor,PCA with MLRand applied it to predict charges and seen the correlation between predicted and actual results.

By this report we conclude the insurance cost of the people by their age ,region ,sex,BMI,and whether they smoke or not.

Overall, machine learning models are very predictive to predict insurance cost. We iteratively developed several predictive models for this .