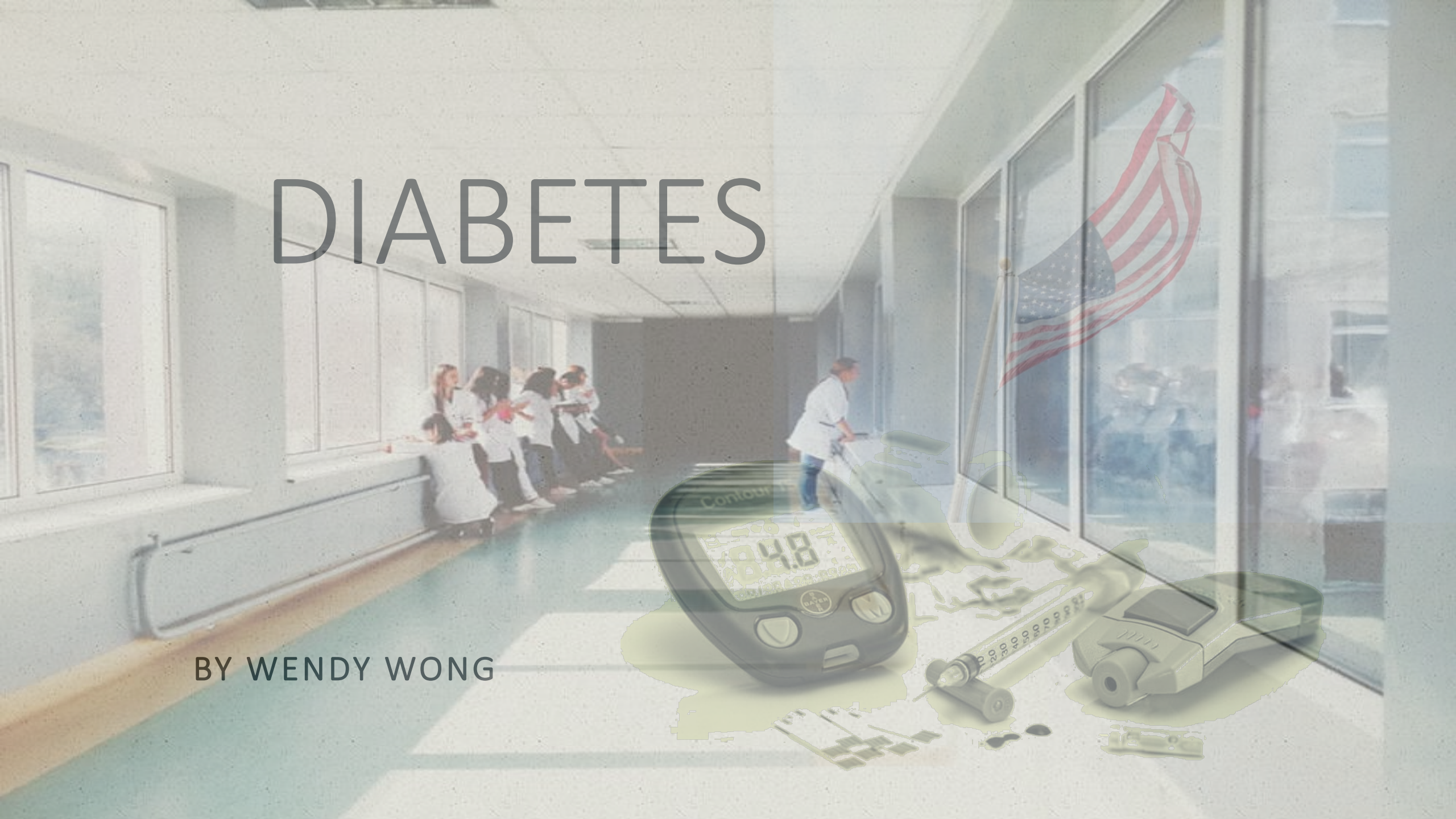
DIABETES



BY WENDY WONG

Data Science Process

1. **Ask an interesting question**
2. **Get the data**
3. **Understand the data**
4. **Prepare the Data**
5. **Explore the Data**
6. **Model the data**
7. **Evaluate the model for success**

Research Problem

What do you want to know?

1. Predict how often do people with diabetes get admitted into Emergency section of the hospital?
2. What are the main reasons for hospital admission is it just for diabetes or another medical condition?
3. What characteristics do diabetes patients all have in common, what are the patterns from their health records?

Further Questions:

1. Predict how many people are re-admitted into hospital who have diabetes ?
2. Use Unsupervised Learning to identify patterns and groups of people who have diabetes
3. Which demographic is likely to be admitted into emergency for having diabetes? Age, race, gender.

NB: this study is dedicated to my father who has diabetes and has multiple medications and diagnoses.

Get the data

I did not have access to explore diabetes data in Australia due to our strict privacy laws for medical data which is not open source.

Data:

- UCI Machine Learning Repository

https://archive.ics.uci.edu/ml/datasets/Diabetes+130-US+hospitals+for+years+1999-2008

- The data was donated on 5 March 2014 by the Center for Clinical and Translational Research, Virginia Commonwealth University,
- This data is a de-identified abstract of the Health Facts database
- The dataset represents 10 years (1999-2008) of clinical care at 130 US hospitals and integrated delivery networks
- (1) It is an inpatient encounter (a hospital admission).
  (2) It is a diabetic encounter, that is, one during which any kind of diabetes was entered to the system as a diagnosis.

The data contains such attributes as **patient number, race, gender, age, admission type, time in hospital, medical specialty of admitting physician, number of lab test performed, HbA1c test result, diagnosis, number of medication, diabetic medications, number of outpatient, inpatient, and emergency visits** in the year before the hospitalization, etc.

- Multivariate data: 50 variables, 100,000 records both electronically recorded and hand-written

Citation:

Beata Strack, Jonathan P. DeShazo, Chris Gennings, Juan L. Olmo, Sebastian Ventura, Krzysztof J. Cios, and John N. Clore, “Impact of HbA1c Measurement on Hospital Readmission Rates: Analysis of 70,000 Clinical Database Patient Records,” BioMed Research International, vol. 2014, Article ID 781670, 11 pages, 2014.

Data Pre-processing

Of the **50 variables** most were medical terms that I did not require for the analysis do I dropped the variables and created a dataframe of 'columns of interest' to answer my quetstions and extracted the **16 variables:**

- **age, race, gender, diabetesMed, insulin, num_medications, number_diagnoses, admission_type_id, admission_source_id, time_in_hospital, number_procedures, num_lab_procedures, number_outpatient, number_inpatient, readmitted,number_emergency**

- Categorical records where converted and encoded into labels or classes before the data was ready for exploration and machine learning

101756	443842070	140199494	Other	Female	[60-70)	?	1	1	7
101757	443842136	181593374	Caucasian	Female	[70-80)	?	1	1	7
101758	443842340	120975314	Caucasian	Female	[80-90)	?	1	1	7
101759	443842778	86472243	Caucasian	Male	[80-90)	?	1	1	7
101760	443847176	50375628	AfricanAmerican	Female	[60-70)	?	1	1	7
101761	443847548	100162476	AfricanAmerican	Male	[70-80)	?	1	3	7
101762	443847782	74694222	AfricanAmerican	Female	[80-90)	?	1	4	5
101763	443854148	41088789	Caucasian	Male	[70-80)	?	1	1	7
101764	443857166	31693671	Caucasian	Female	[80-90)	?	2	3	7
101765	443867222	175429310	Caucasian	Male	[70-80)	?	1	1	7

101766 rows × 50 columns

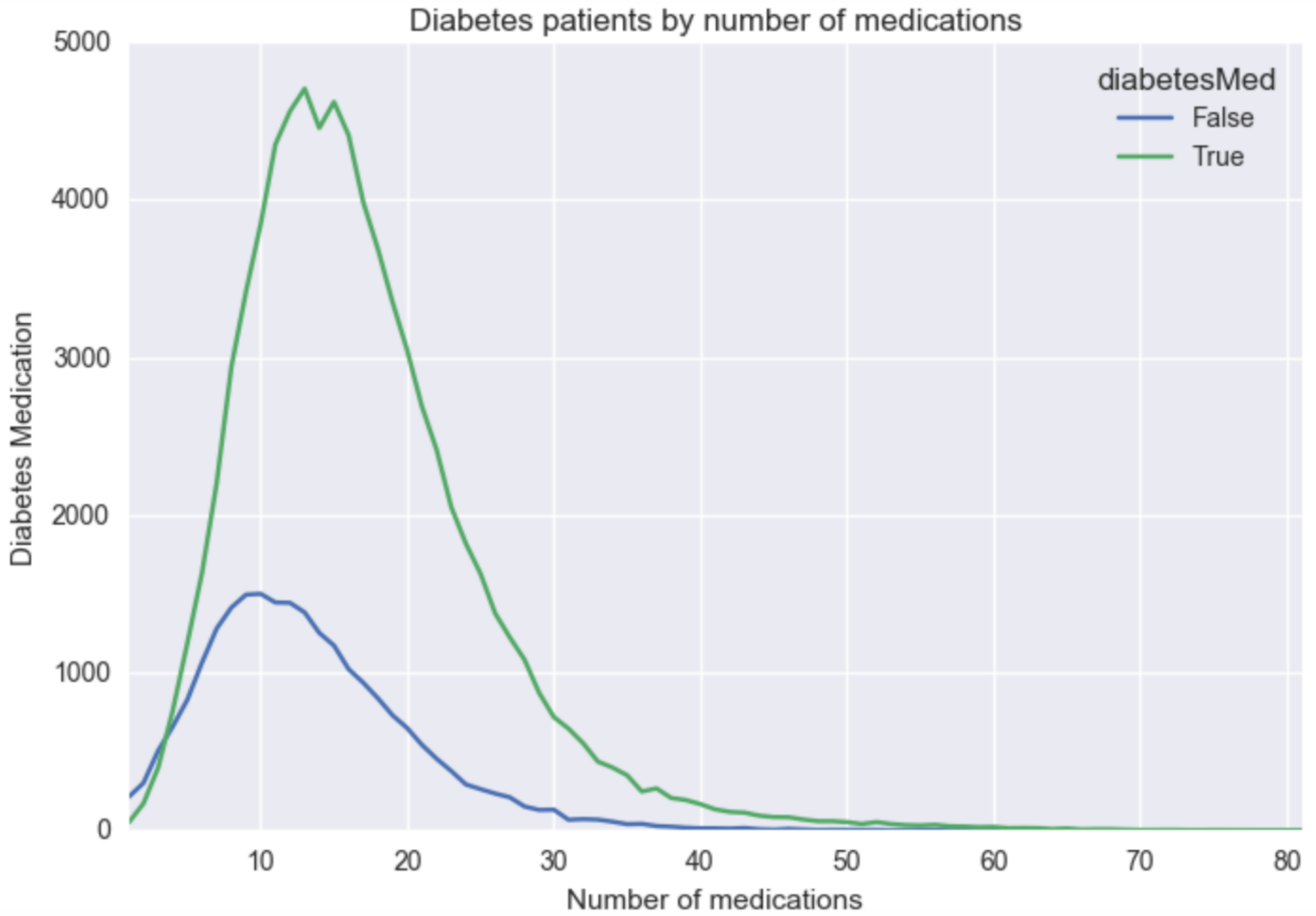
Data Pre-processing

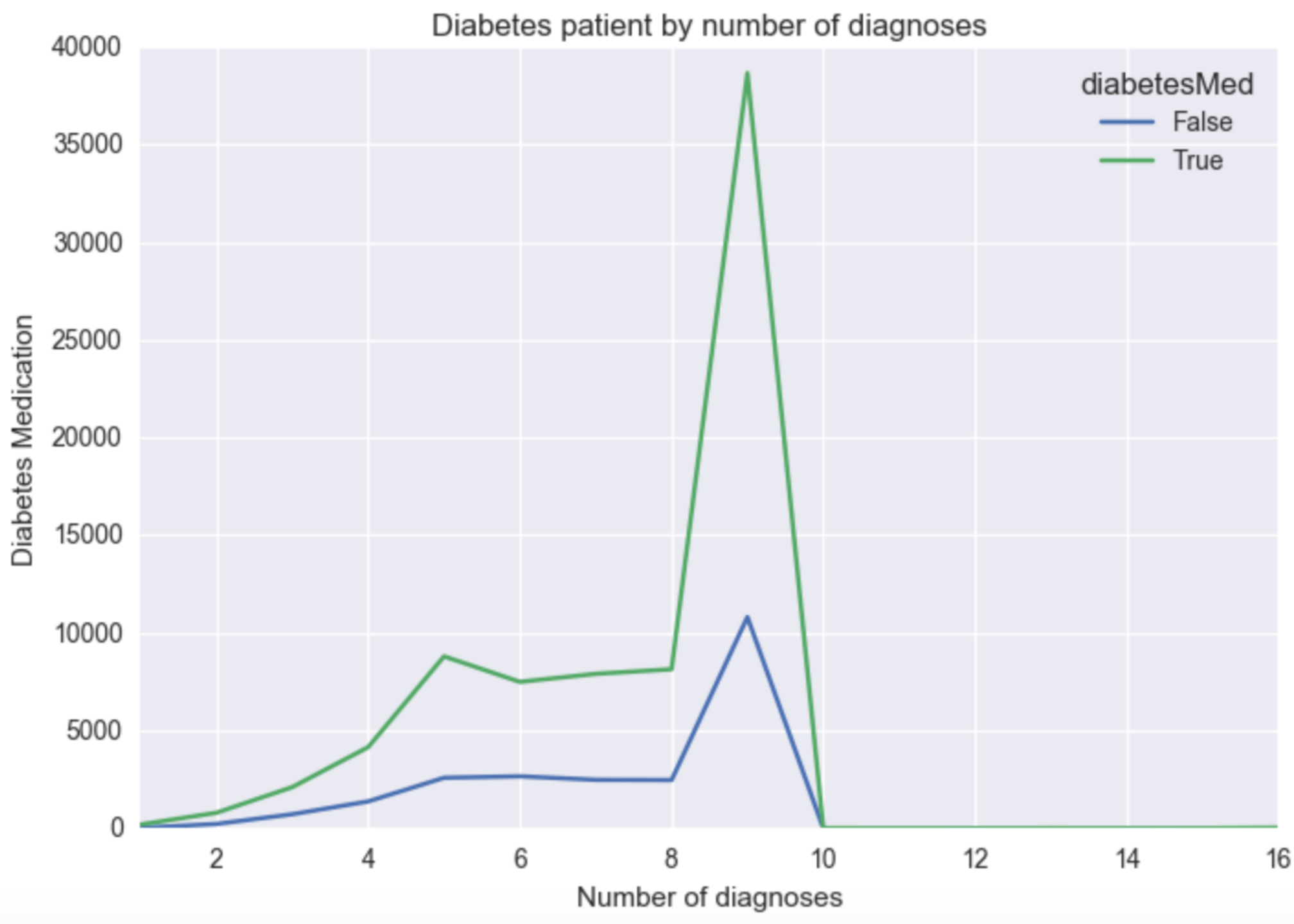
Clean and Transform the data :

80% of time, removed duplicates, missing records, dropped variables, encoded categorical variables and created new dataframe on 'cleaned data'

```
: # Use Scikit-learn estimators for classification convert labels to integers internally
  # we enumerate the class labels starting at 0:

  import numpy as np
  race_mapping = {label:idx for idx,label in
                  enumerate(np.unique(diabetic_data['race']))}
  race_mapping
  {'?':0,'Caucasian':1,'AfricanAmerican':2,'Other':3,'Hispanic':4,'Asian':5}

  gender_mapping = {label:idx for idx,label in
                  enumerate(np.unique(diabetic_data['gender']))}
  gender_mapping
  {'Male':0,'Female':1,'Unknown/Invalid':2}

  age_mapping = {label:idx for idx,label in
                  enumerate(np.unique(diabetic_data['age']))}
  age_mapping
  {'[0-10)':0,'[10-20)':1,'[20-30)':2,'[30-40)':3,'[40-50)':4,'[50-60)':5,'[60-70)':5,'[70-80)':7,'[80-90)':8,'[90-100)':

  insulin_mapping = {label:idx for idx,label in
                  enumerate(np.unique(diabetic_data['insulin']))}
  insulin_mapping
  {'No':0,'Steady':1,'Up':2,'Down':3}

  diabetesMed_mapping = {label:idx for idx,label in
                  enumerate(np.unique(diabetic_data['diabetesMed']))}
  diabetesMed_mapping
  {'No':0,'Yes':1}
```

Explore the data – Non-Linear


Diabetes patients by number of medications
diabetesMed
False
True
Diabetes Medication
Number of medications
0
1000
2000
3000
4000
5000
10
20
30
40
50
60
70
80


Diabetes patient by number of diagnoses
Diabetes Medication
Number of diagnoses
diabetesMed
False
True
40000
35000
30000
25000
20000
15000
10000
5000
0
2
4
6
8
10
12
14
16

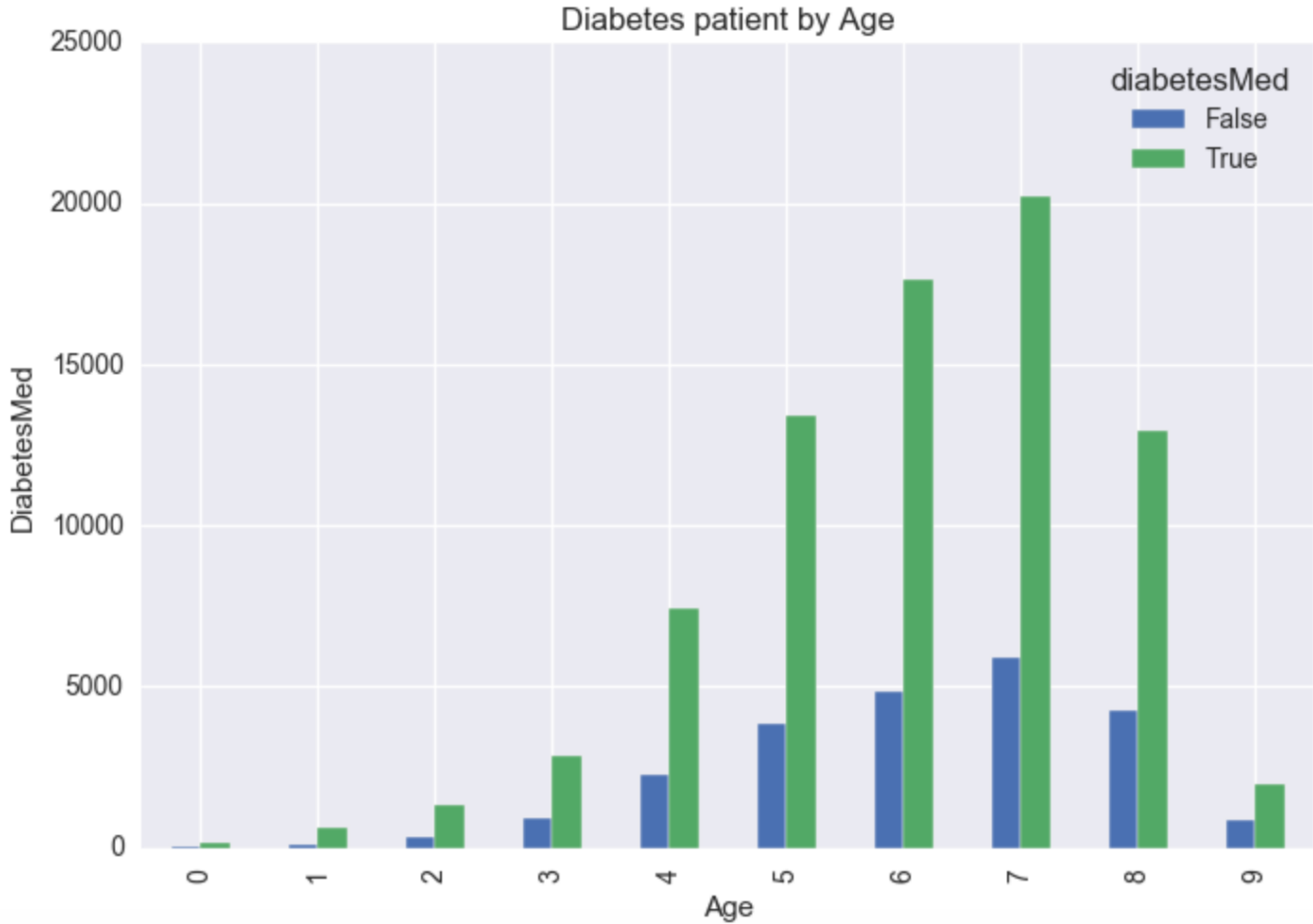
Correlation – relationship between variables

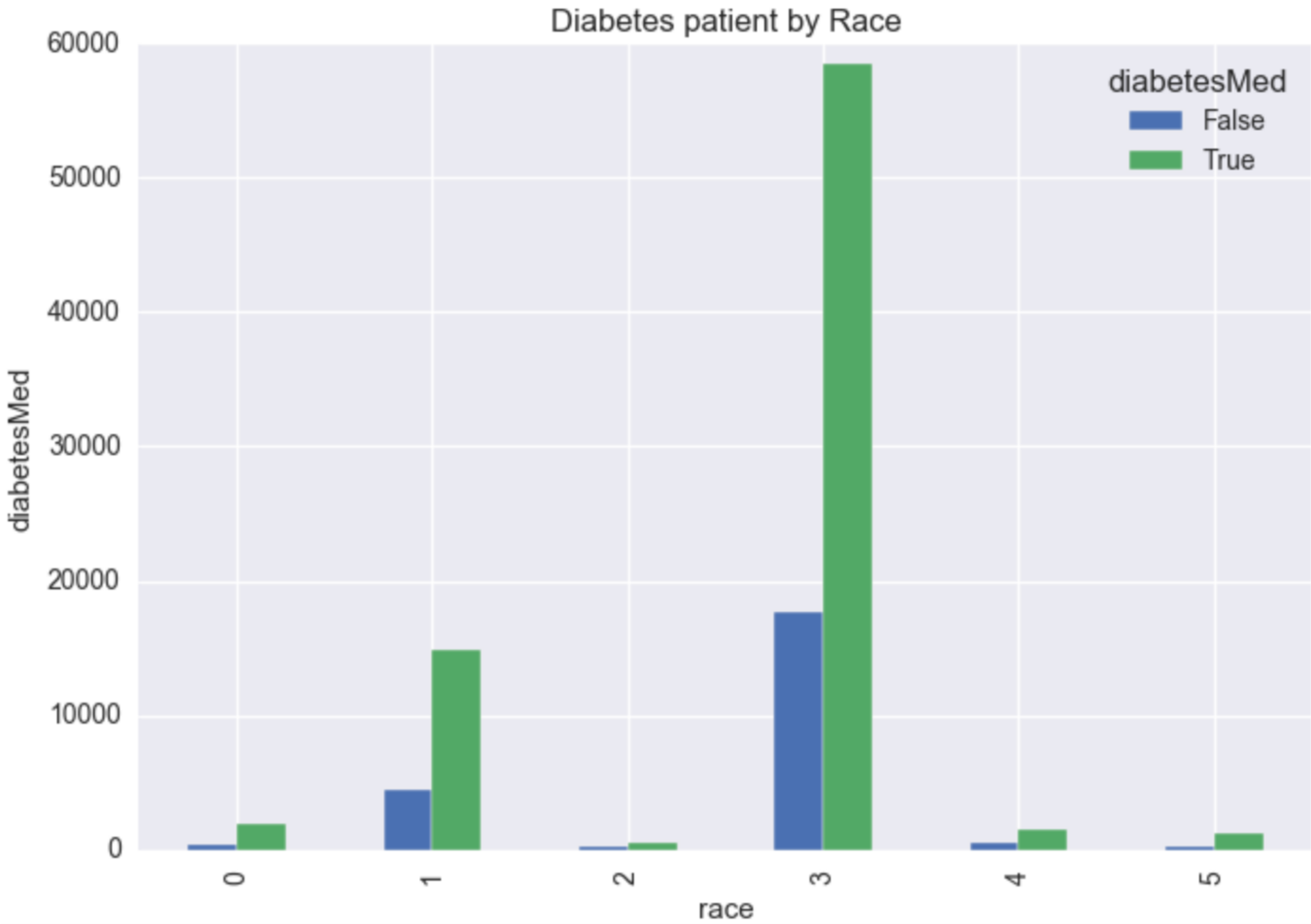
- num_medications vs diabetesMed = 0.18
- number_diagnoses vs diabetesMed = 0.02

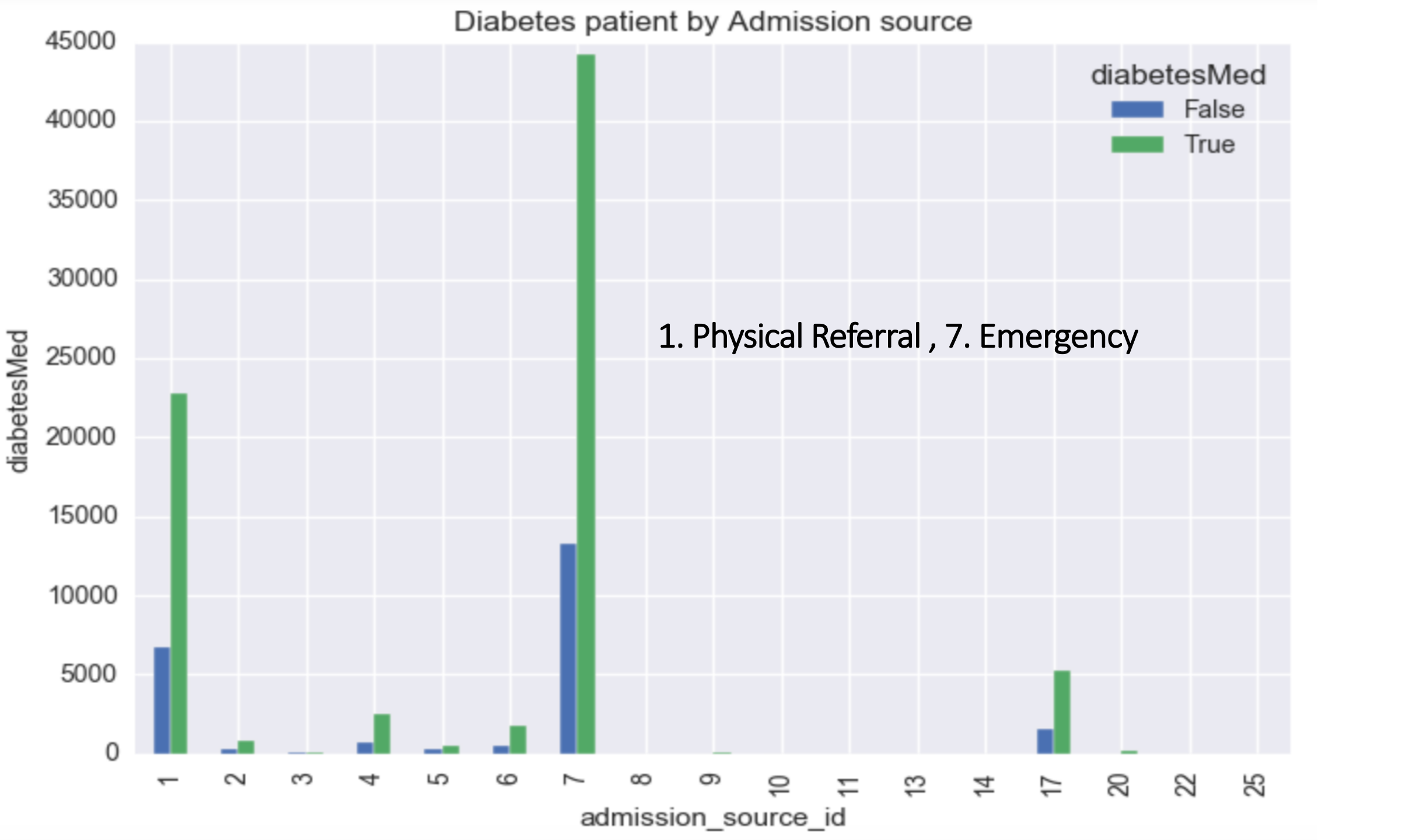
	admission_type_id	number_diagnoses	num_procedures	admission_source_id	readmitted	diabetesMed	race	number_
admission_type_id	1.000000	-0.117126	0.129888	0.106654	0.008950	-0.000310	0.098659	0.026511
number_diagnoses	-0.117126	1.000000	0.073734	0.072114	-0.104820	0.021186	0.084176	0.094152
num_procedures	0.129888	0.073734	1.000000	-0.135400	0.038235	-0.006821	0.027087	-0.02481
admission_source_id	0.106654	0.072114	-0.135400	1.000000	-0.031816	0.001500	0.031173	0.027244
readmitted	0.008950	-0.104820	0.038235	-0.031816	1.000000	-0.057306	-0.015184	-0.06855
diabetesMed	-0.000310	0.021186	-0.006821	0.001500	-0.057306	1.000000	-0.006367	0.016456
race	0.098659	0.084176	0.027087	0.031173	-0.015184	-0.006367	1.000000	0.048456
number_outpatient	0.026511	0.094152	-0.024819	0.027244	-0.068552	0.016456	0.048456	1.000000
num_lab_procedures	-0.143713	0.152773	0.058066	0.048885	-0.037976	0.033107	-0.023033	-0.00760
gender	0.014592	-0.003407	0.059980	-0.003843	0.014533	0.015901	0.055271	-0.01148
age	-0.007209	0.242597	-0.030104	0.044696	-0.030271	-0.022601	0.114684	0.023724
num_medications	0.079535	0.261526	0.385767	-0.054533	-0.051772	0.186910	0.027935	0.045197
number_inpatient	-0.038161	0.104710	-0.066236	0.036314	-0.234283	0.026001	-0.006569	0.107338
number_emergency	-0.019116	0.055539	-0.038179	0.059892	-0.103024	0.025923	-0.012629	0.091459
time_in_hospital	-0.012500	0.220186	0.191472	-0.006965	-0.057718	0.062520	-0.015324	-0.00891
insulin	-0.007858	0.026834	0.009091	-0.010920	-0.003113	0.264170	-0.029865	0.013360

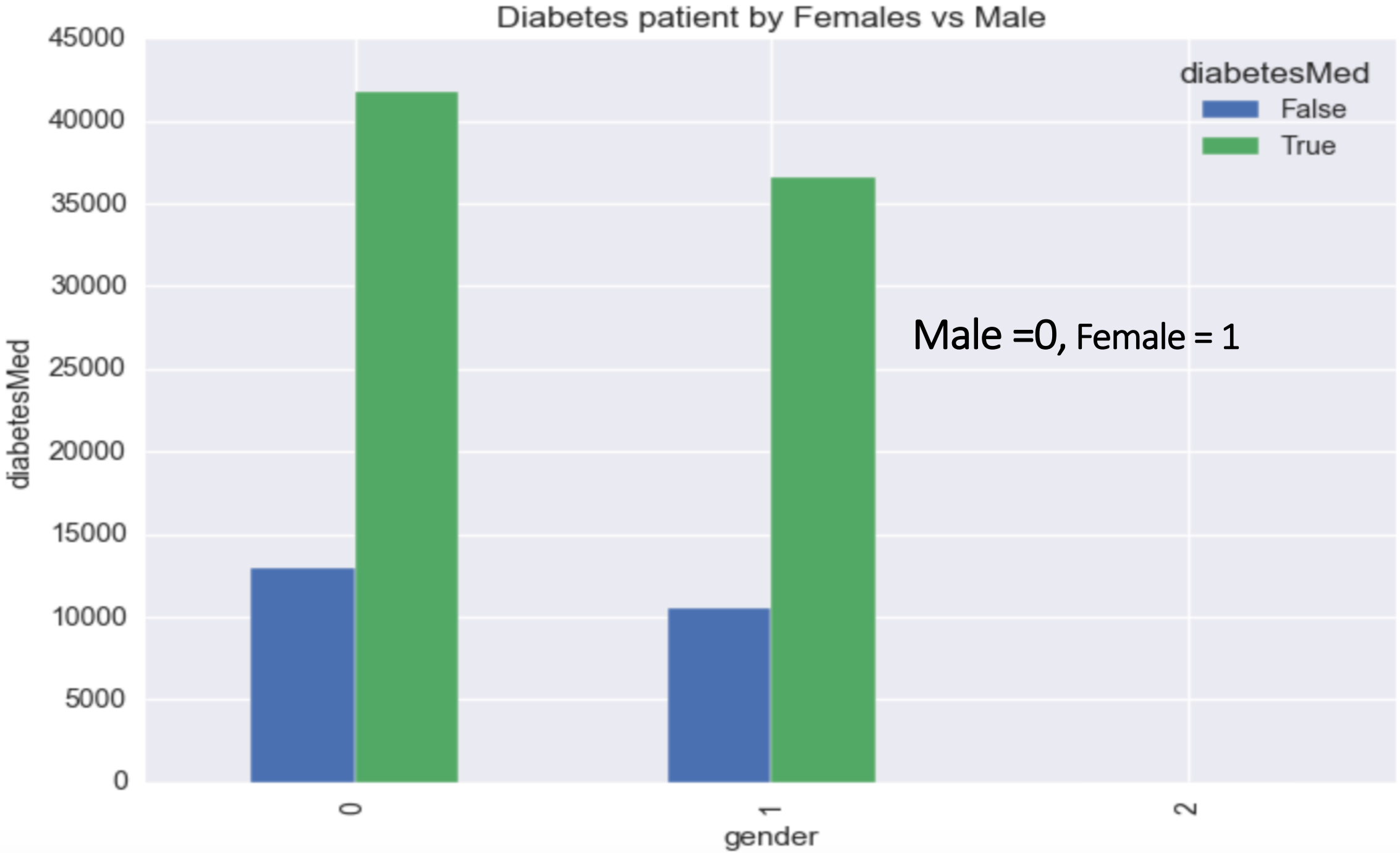
Correlation – relationship between variables

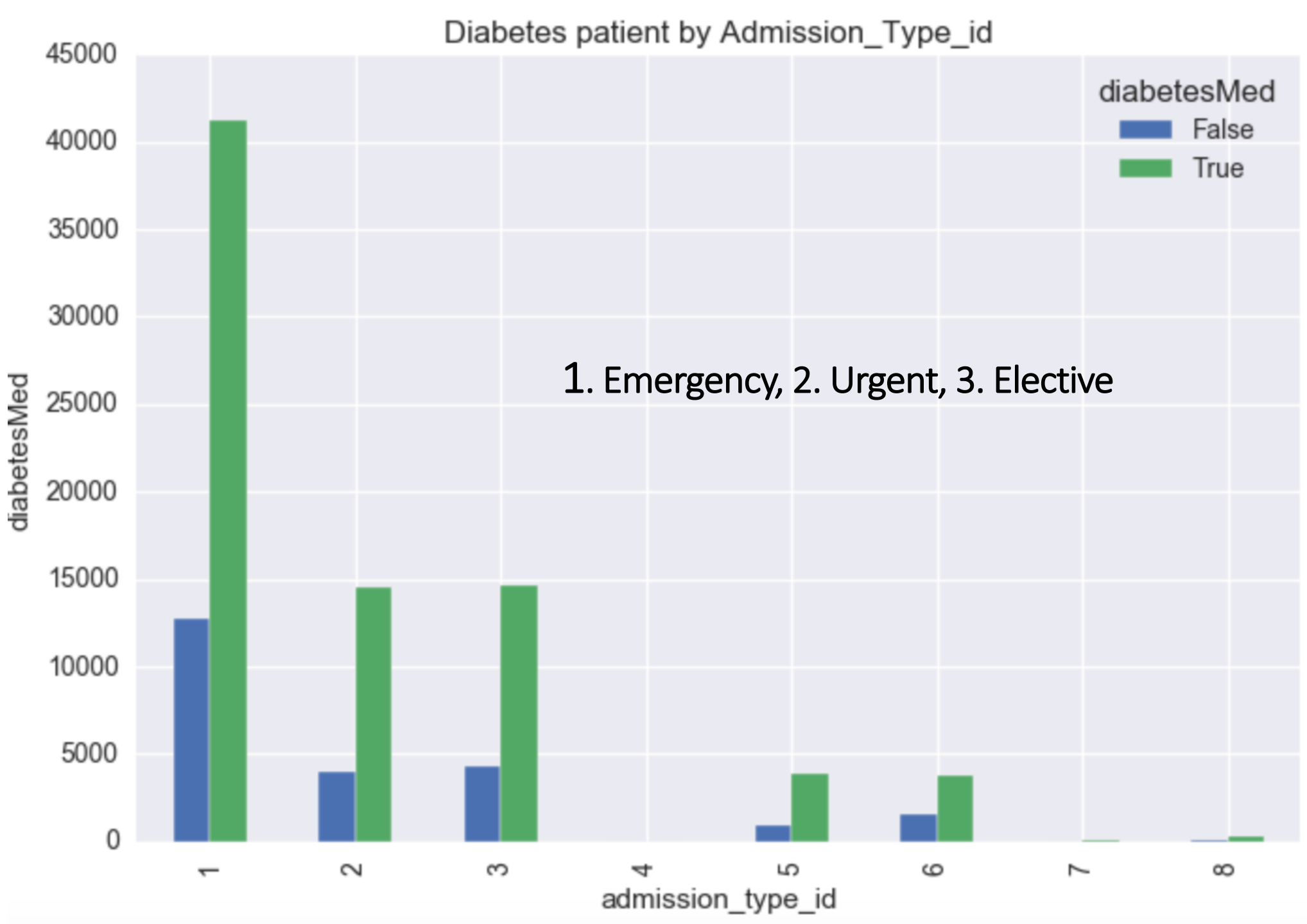
- Number_diagnoses vs age = 0.24
- number_diagnoses vs number diagnoses = 0.2615

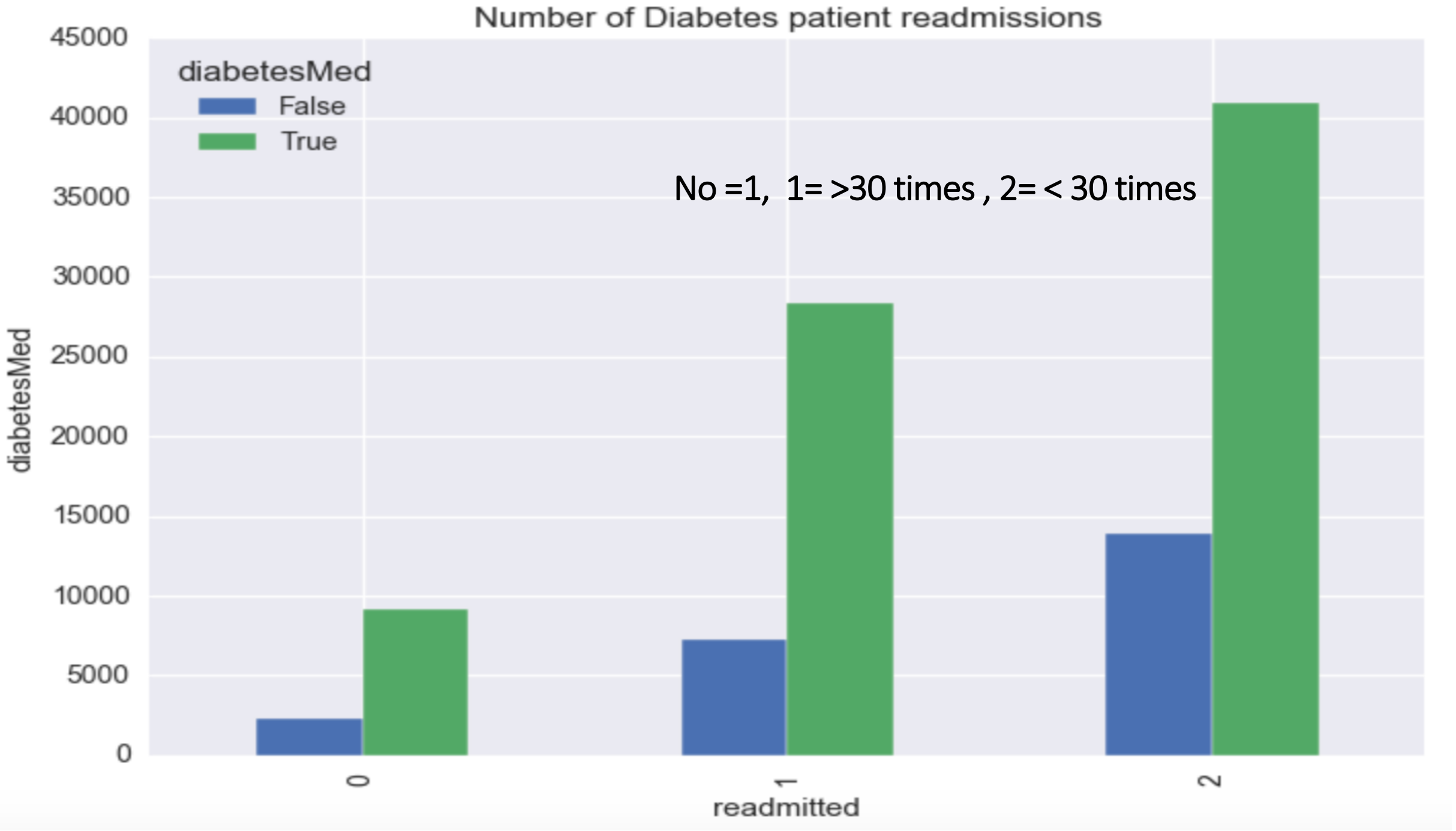

Diabetes patient by Age
diabetesMed
False
True
DiabetesMed
25000
20000
15000
10000
5000
0
0
1
2
3
4
5
6
7
8
9
Age


Diabetes patient by Race
diabetesMed
False
True
60000
50000
40000
30000
20000
10000
0
diabetesMed
0
1
2
3
4
5
race


Diabetes patient by Admission source
diabetesMed
False
True
1. Physical Referral , 7. Emergency
diabetesMed
admission_source_id
45000
40000
35000
30000
25000
20000
15000
10000
5000
0
1
2
3
4
5
6
7
8
9
10
11
13
14
17
20
22
25


Diabetes patient by Females vs Male
diabetesMed
False
True
45000
40000
35000
30000
25000
20000
15000
10000
5000
0
diabetesMed
0
1
2
gender
Male =0, Female = 1


Diabetes patient by Admission_Type_id
diabetesMed
False
True
45000
40000
35000
30000
25000
20000
15000
10000
5000
0
diabetesMed
1. Emergency, 2. Urgent, 3. Elective
1
2
3
4
5
6
7
8
admission_type_id


Number of Diabetes patient readmissions
diabetesMed
False
True
No =1, 1= >30 times , 2= < 30 times
45000
40000
35000
30000
25000
20000
15000
10000
5000
0
diabetesMed
0
1
2
readmitted

Describe – Diabetic patients

		admission_source_id	admission_type_id	age	gender	insulin	num_lab_procedures	num_medications
diabetesMed								
0	count	23403.000000	23403.000000	23403.000000	23403.000000	23403.000000	23403.000000	23403.000000
	mean	5.743281	2.024826	6.162629	0.447934	1.000000	41.903730	13.242063
	std	4.027094	1.477064	1.602481	0.497378	0.000000	19.092522	7.030729
	min	1.000000	1.000000	0.000000	0.000000	1.000000	1.000000	1.000000
	25%	1.000000	1.000000	5.000000	0.000000	1.000000	31.000000	8.000000
	50%	7.000000	1.000000	6.000000	0.000000	1.000000	43.000000	12.000000
	75%	7.000000	3.000000	7.000000	1.000000	1.000000	55.000000	17.000000
	max	20.000000	8.000000	9.000000	2.000000	1.000000	129.000000	69.000000
1	count	78363.000000	78363.000000	78363.000000	78363.000000	78363.000000	78363.000000	78363.000000
	mean	5.757768	2.023761	6.077013	0.466776	1.526562	43.451603	16.852022
	std	4.075081	1.435821	1.591048	0.498949	0.921925	19.831052	8.247816
	min	1.000000	1.000000	0.000000	0.000000	0.000000	1.000000	1.000000
	25%	1.000000	1.000000	5.000000	0.000000	1.000000	32.000000	11.000000
	50%	7.000000	1.000000	6.000000	0.000000	2.000000	45.000000	15.000000
	75%	7.000000	3.000000	7.000000	1.000000	2.000000	57.000000	21.000000
	max	25.000000	8.000000	9.000000	2.000000	3.000000	132.000000	81.000000

1. Linear Regression: predict y= categorical response ' readmitted ', X = 'diabetesMed'

```
print('R2 score: %.3f' % lr.score(X_test, y_test))
```

```
R2 score: 0.003
```

R-squared : 0.003 or 0%

The R-squared does not represent a robust model because the response is not a continous variable but is a categorical variable 'readmitted' hence the variance within the model is not explained by a single variable 'diabeteMed' which is also a categorical variable.

Hence, Multi-momial Logistic regression should be applied to fit and predict a categorical outcome with three classes for 'readmitted'.

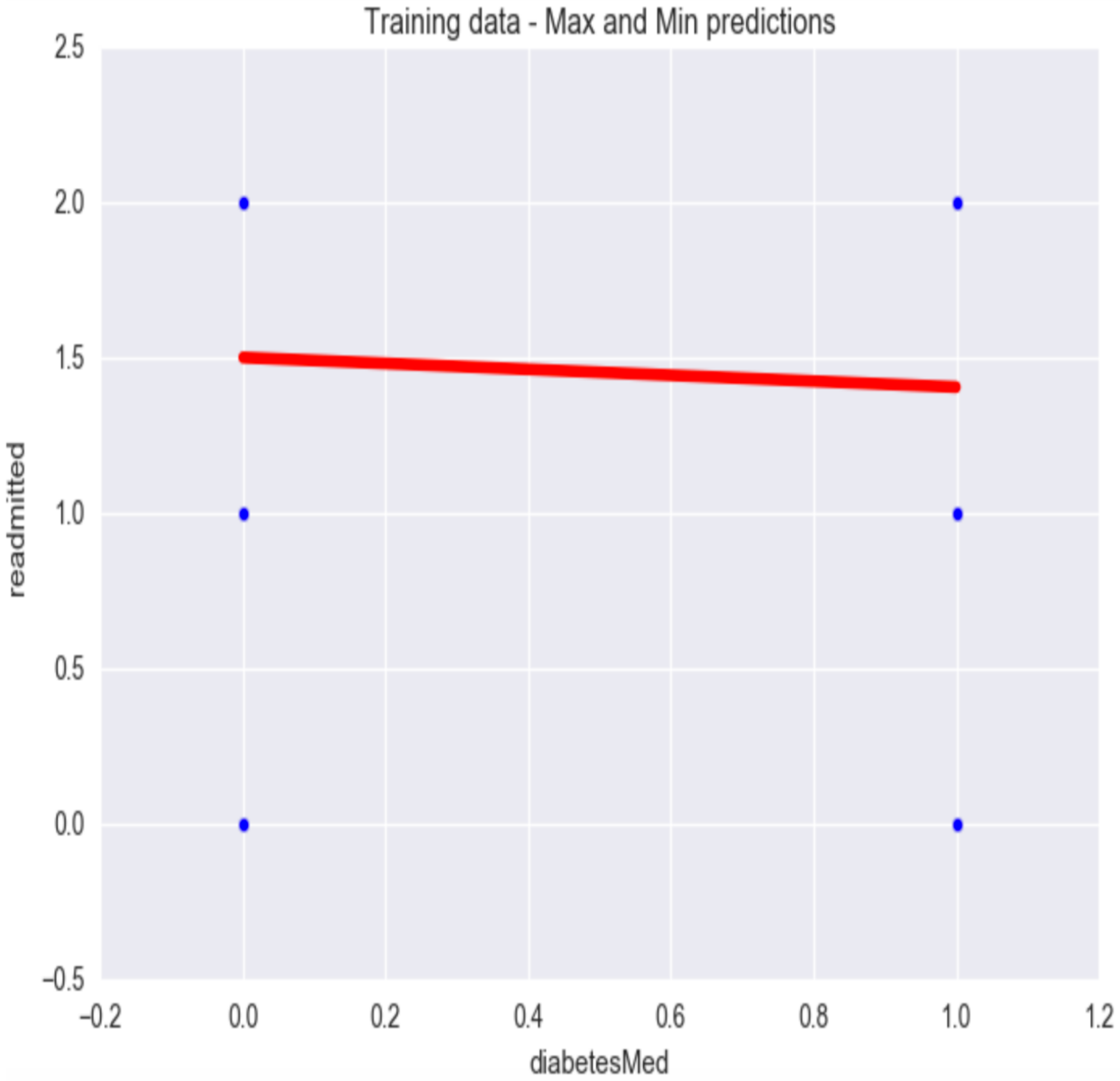
```
lr.coef_
```

```
array([-0.09514219])
```

*Interpretation of coefficient: A 1 unit increase in 'readmitted' will lead to a a decrease in 'diabetesMed' by 0.095142 units with all other variables being unchanged in the model.The coefficient is negative because the weight or beta is a binary variable.That is 'diabetesMed' is 0 (for no medication) or 1(takes medication)

```
lr.intercept_
```

```
1.5019830374031344
```



OLS Regression Results

Dep. Variable:	readmitted	**R-squared:**	0.003
Model:	OLS	**Adj. R-squared:**	0.003
Method:	Least Squares	**F-statistic:**	335.3
Date:	Sun, 11 Dec 2016	**Prob (F-statistic):**	8.95e-75
Time:	12:49:18	**Log-Likelihood:**	-1.0559e+05
No. Observations:	101766	**AIC:**	2.112e+05
Df Residuals:	101764	**BIC:**	2.112e+05
Df Model:	1		
Covariance Type:	nonrobust		

	coef	**std err**	**t**	**P>|t|**	**[95.0% Conf. Int.]**
Intercept	1.4993	0.004	335.833	0.000	1.491 1.508
diabetesMed	-0.0932	0.005	-18.311	0.000	-0.103 -0.083

Omnibus:	10809.731	**Durbin-Watson:**	1.976
Prob(Omnibus):	0.000	**Jarque-Bera (JB):**	11690.094
Skew:	-0.783	**Prob(JB):**	0.00
Kurtosis:	2.446	**Cond. No.**	3.95

2. Classification (logistic): predict y= categorical response ' readmitted ', 1 feature

Train/Test split :

Scikit-Learn randomly split the training and test set for fitting a logistic regression model.

- Feature columns (preds) = 'diabetesMed'

Coefficients :

- ' diabetesMed' = 0.214549954864775O4

Model Accuracy on Prediction:

0.53911915571 – **1 feature**

- Sensitivity (Recall and True positive rate) = TP/float(TP+FN) = 54864/0+54864=1.0
- Specificity (precision) = FN/FN+TP = 0/float(0+35545) =0.0

Evaulate Logistic regression prediction via **Confusion Matrix:**

0	0	11357
0	0	35545
0	0	54864

2. Classification (logistic): predict y= categorical response ‘ readmitted ’, 2 features

Train/Test split :

Scikit-Learn randomly split the training and test set for fitting a logistic regression model.

- Feature columns (X_train) = ‘diabetesMed’ and ‘age’

Coefficients :

- ‘ diabetesMed’ = 0.2365835 and
- ‘age’ = 0.036746

Model Accuracy on Test Set:

0.543432 – **2 features**

0.543432 **- 3 features with extra feature** ‘gender’

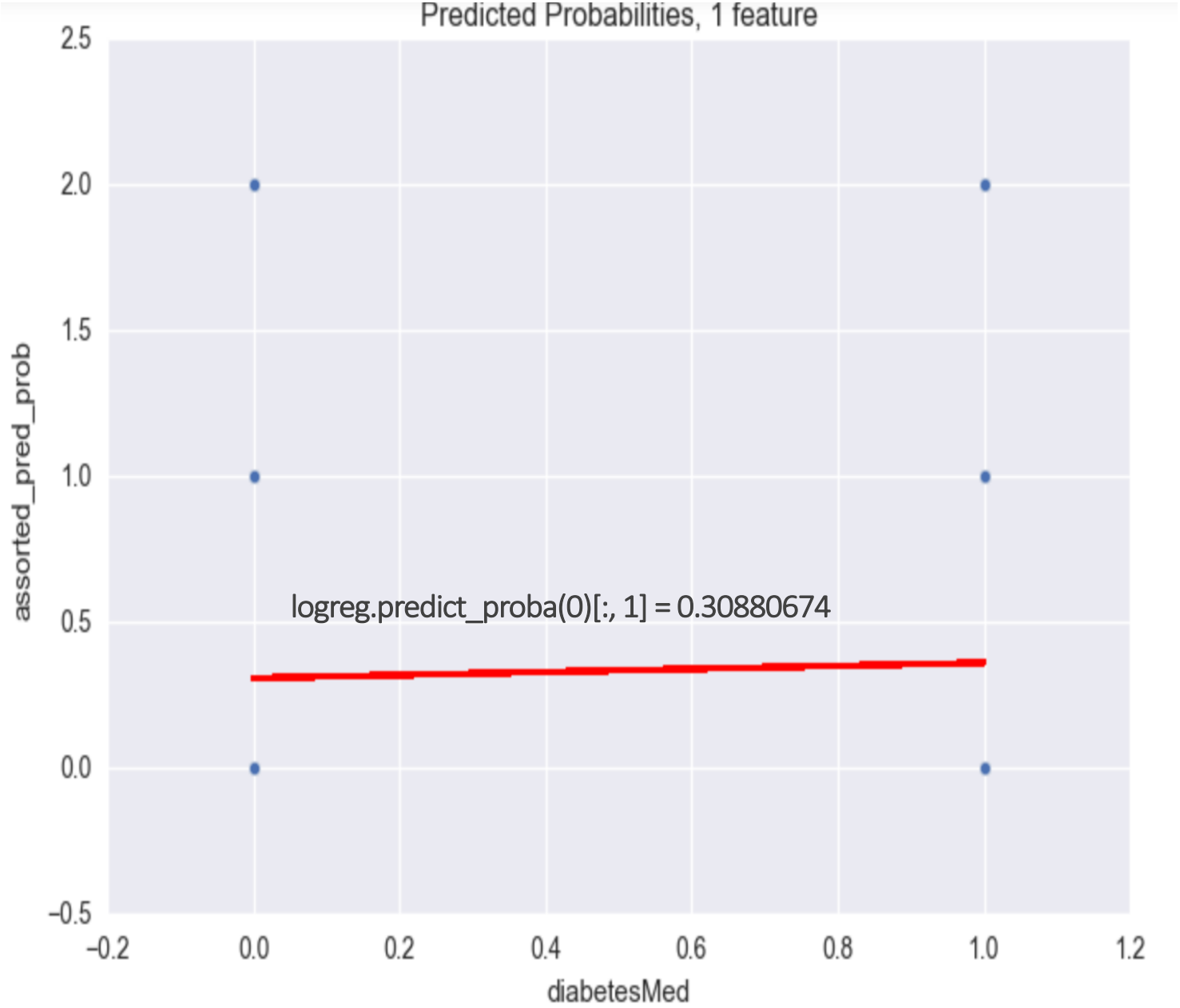
- Sensitivity (Recall and True positive rate) = TP/float(TP+FN) = 8859/0+8859=1.0
- Specificity (precision) = FN/FN+TP = 0/float(0+2757) =0.0

Evaulate Logistic regression prediction via **Confusion Matrix:**

0	0	2757
0	0	8859
0	0	13826

Interpretation of intercept: For an 'diabetesMed' value of 0, the log-odds of 'assorted' is -2.24

Interpretation of Coefficient: A 1 unit increase in 'diabetesMed' is associated with a 0.214549 unit increase in the log-odds of 'assorted'.



	probability	odds	logodds
0	0.10	0.111111	-2.197225
1	0.20	0.250000	-1.386294
2	0.25	0.333333	-1.098612
3	0.50	1.000000	0.000000
4	0.60	1.500000	0.405465
5	0.80	4.000000	1.386294
6	0.90	9.000000	2.197225
7	0.99	99.000000	4.595120

2. Classification (Logistic): predict y= categorical response ‘ readmitted ’, 6 features

Train/Test split :

Scikit-Learn randomly split the training and test set for fitting a logistic regression model.

- Feature columns (preds) = 'diabetesMed','race','age', 'num_medications','number_diagnoses','gender’

Coefficients :

- ‘ diabetesMed’ = 0.214549954864775O4

Model Accuracy on Prediction:

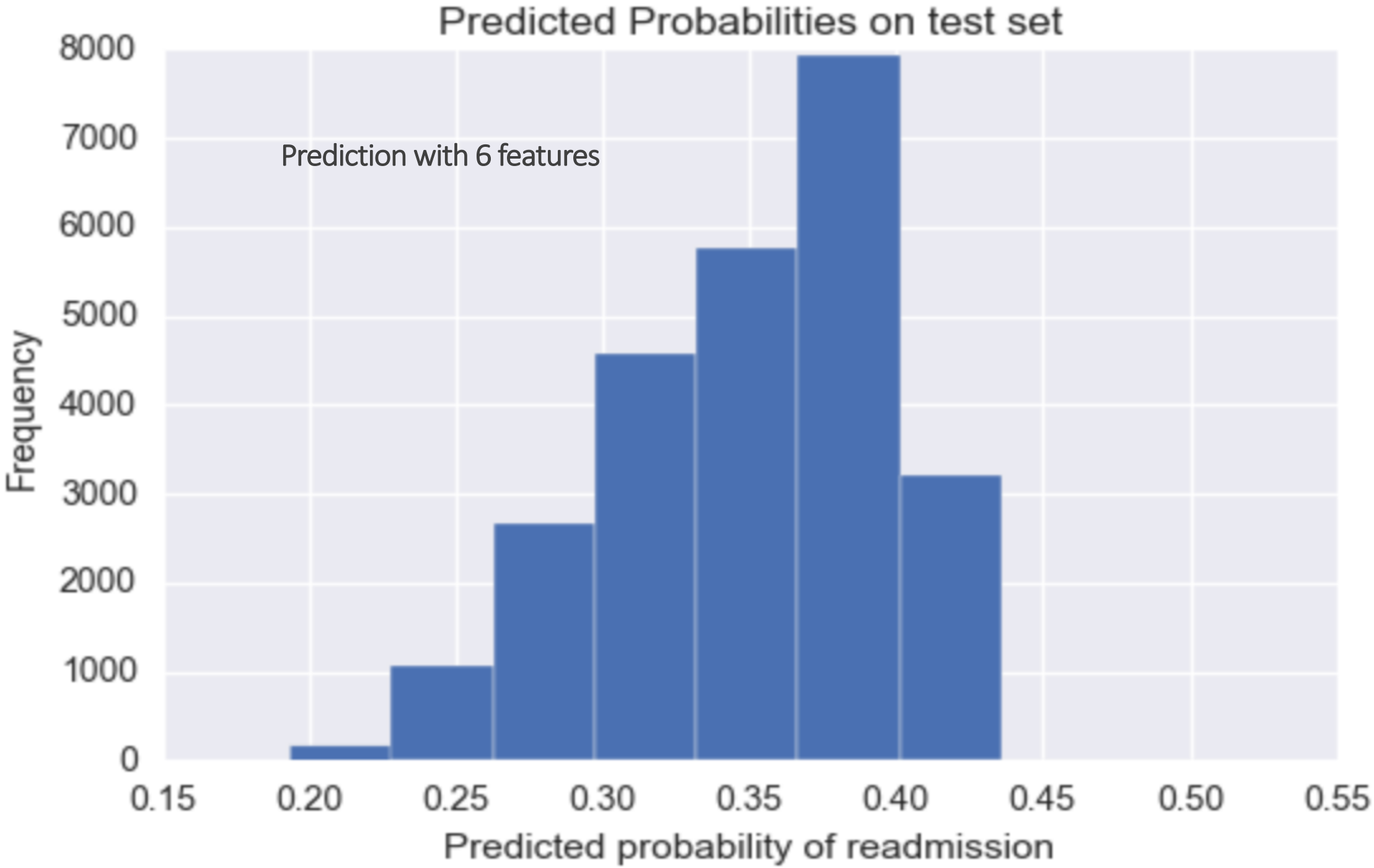
0.543039069256 – **5 feature**

0.543039069256 – **6 features**

Evaulate **6 feature** prediction via **Confusion Matrix:**

0	9	2748
0	4	8855
0	14	13812

- Sensitivity (Recall and True positive rate) = TP/float(TP+FN) = 8855/4+8855= 0.999548481769951 5
- Specificity (precision) = FN/FN+TP = 0/float(9+2748) = 0.0032644178454842 22


Predicted Probabilities on test set
Prediction with 6 features
Frequency
8000
7000
6000
5000
4000
3000
2000
1000
0
0.15
0.20
0.25
0.30
0.35
0.40
0.45
0.50
0.55
Predicted probability of readmission

Model Evaluation: Logistic regression + Cross-Validation (score accuracy)

1.Evaluate the model using 10-fold cross-validation.

Starting with 50 predictors and 20 samples, find 100 predictors

Mean score(Average error or Test Error) = 0.538952215962

2 .Evaluate the model using 5-fold cross-validation:

Starting with 5 predictors and 5 samples, find 25 predictors

Mean score(Average error or Test Error) = 0.538952048149

3. Principal Component Analysis (PCA)

PCA was not conducted because the variables in the diabetes dataset were not all continuous but included categorical and binary predictors.
Hence standardization or normalisation would not explain the variance in categorical variables

4. Shrinkage - Regularization

Select a tuning parameter **alpha** with a **small value** for Lasso regression so that it does not a penalty on the coefficients in the model.

*** Increase the size of alpha so that it penalises the coefficients and **shrinks them towards zero** (Ridge regression coefficients approach zero with penalty term).

- **Lasso regression,** alpha = 0.0001

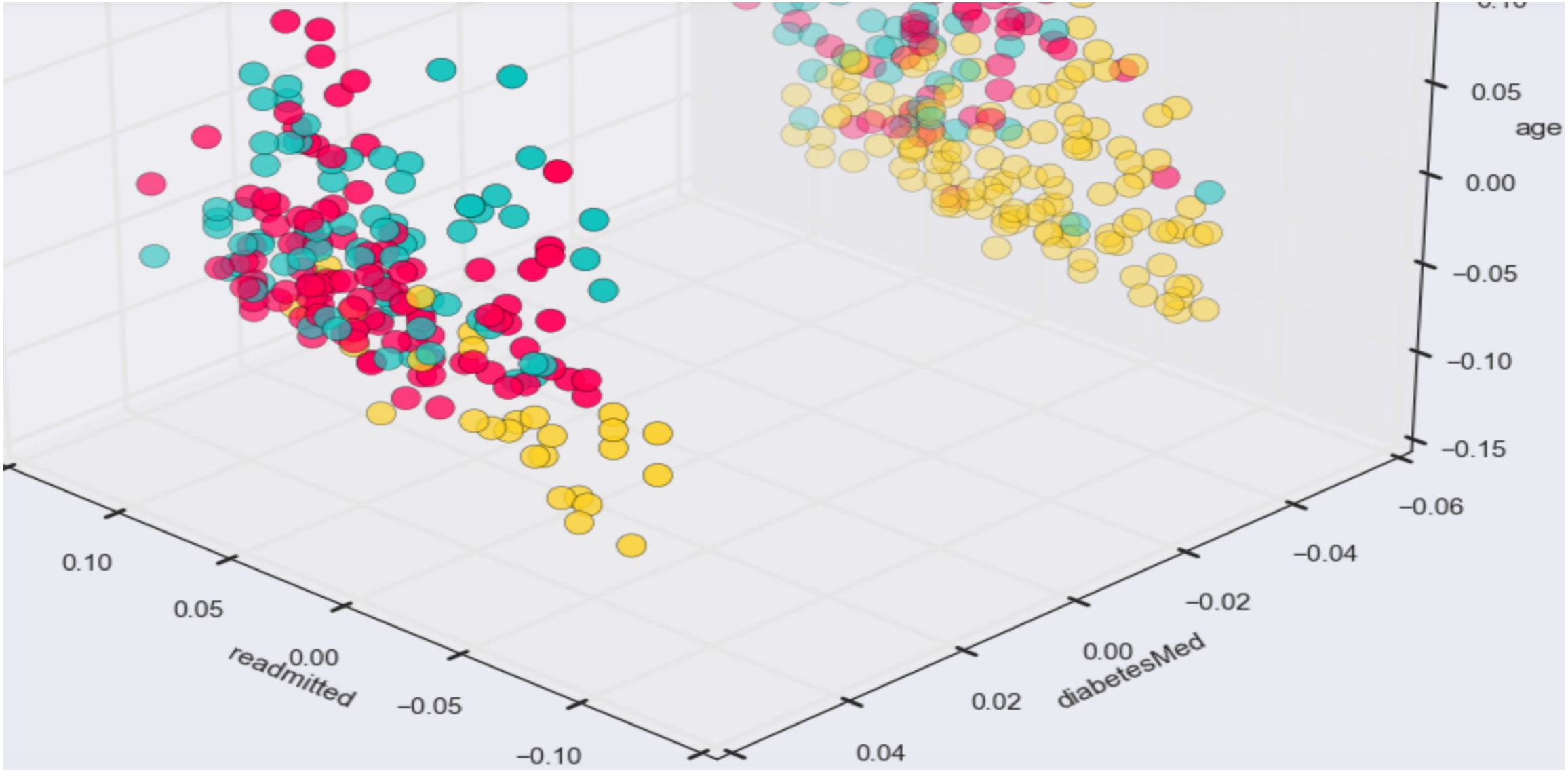
RMSE (Lasso reg.) = 0.806867198652

- **Lasso regression (with cross-validation),** alpha = 0.01 *

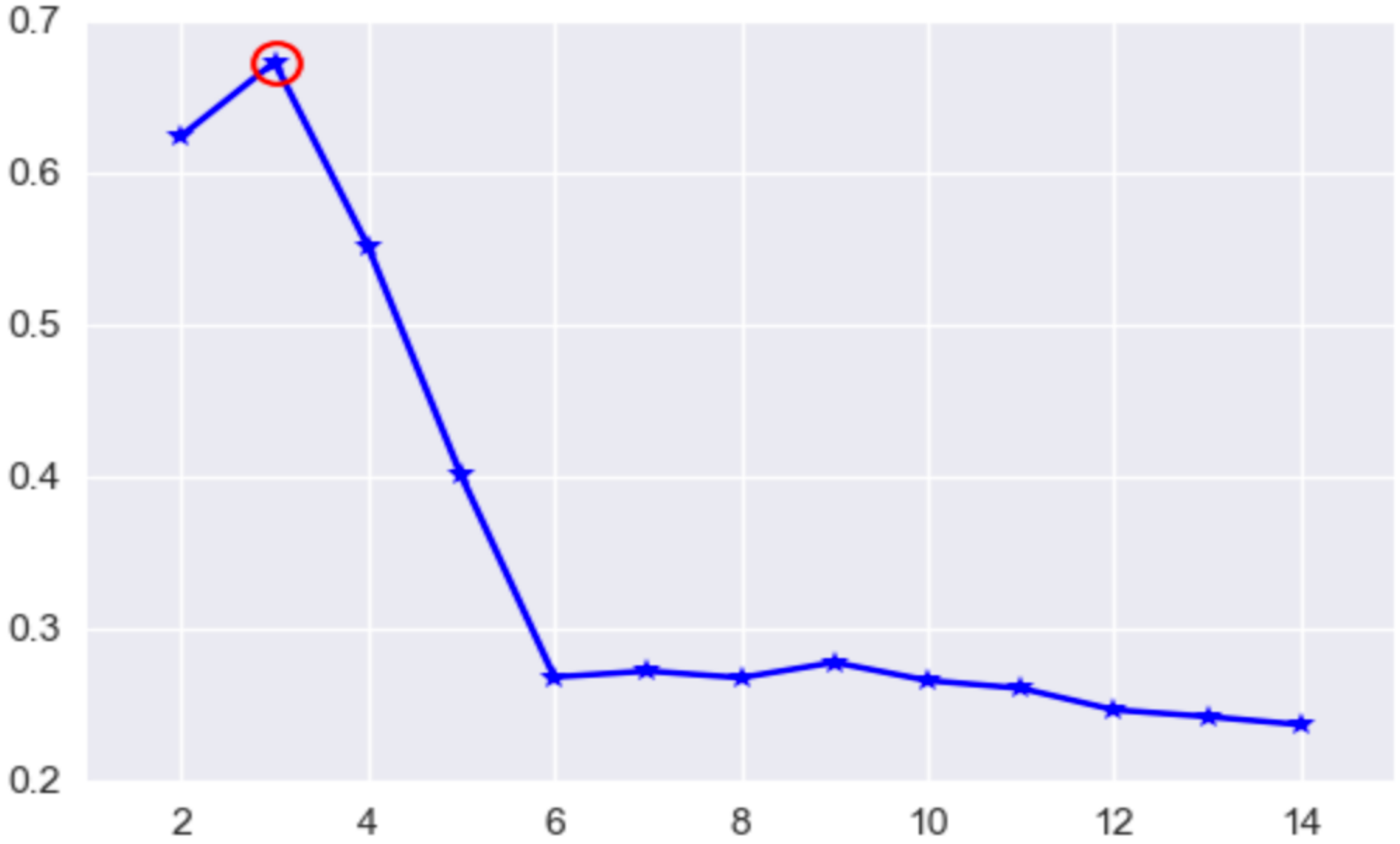
RMSE (Lasso reg.) = 0.806867198652

```
array([ -0.00000000e+00,   0.00000000e+00,   0.00000000e+00,
        -0.00000000e+00,   0.00000000e+00,   4.60882045e-01,
        -0.00000000e+00,   0.00000000e+00,   0.00000000e+00,
         0.00000000e+00,  -0.00000000e+00,   1.17620952e-04,
         0.00000000e+00,   0.00000000e+00,   0.00000000e+00])
```

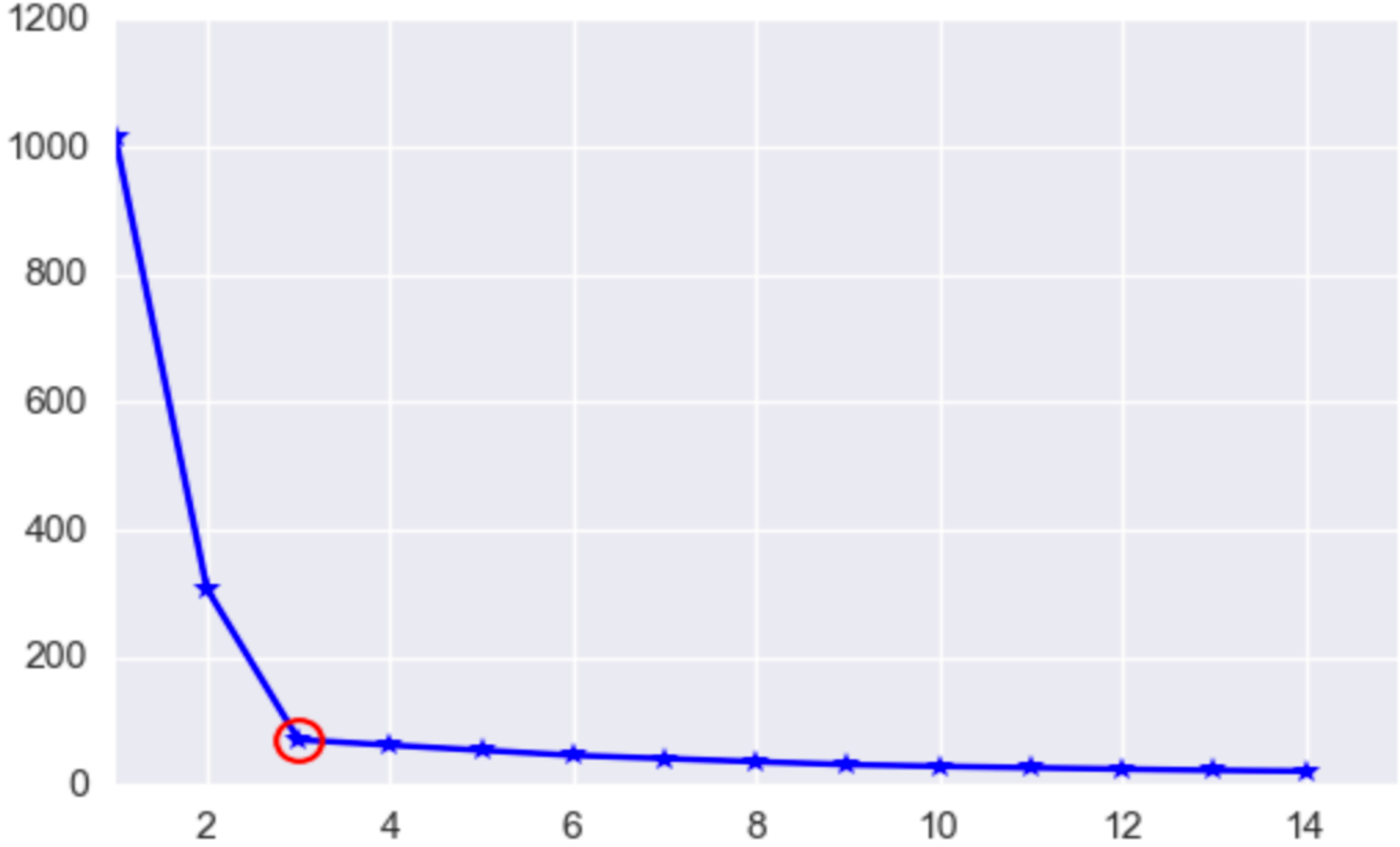
5. Cluster Analysis – computation problems!, variables are high dimensional and limitations of different scales. Could not compute centroids or extract info on one cluster



5. Cluster Analysis – Silhouette Coefficient



5. Cluster Analysis – minimise WSS and minimise K



6. Decision Tree – Feature Importance

- **Easy to interpret**
- Problem of overfitting
- Low Gini Coefficient is a bad split
- **Non-linear data**: scans and splits on a feature that produces the greatest separation between classes in the resulting nodes.
- Classification tree: predict that each observation belongs to the most commonly occurring class of training observations select features that would predict - readmissions for diabetes patients.

6. Decision Tree – 2 Key Insights

- 1.(80-90), 'Other' race group, female will be readmitted for diabetes 4450
- 2.(90-100), 'Other' race group, female will be readmitted for diabetes 501

- 1. 132: age (10 to 20) female, 'other' race group will be readmitted into hospital as an emergency who is a diabetes patient
- 2. 32: age (0to 10) female, 'other' race group will be readmitted into hospital as an emergency patient with diabetes
- 3. 41: age (10 to 20) male, 'other' race group will be readmitted into hospital as an emergency patient with diabetes

6. Decision Tree – Feature Importance by rank

	0	1
6	number_diagnoses	0.738717
3	diabetesMed	0.196054
0	admission_type_id	0.065229
1	gender	0.000000
2	age	0.000000
4	race	0.000000
5	num_medications	0.000000
7	time_in_hospital	0.000000

Model accuracy:

0.5434321201163430 1

Confusion Matrix:

predicted	2
actual	
0	2757
1	8859
2	13826

7. Ensemble – Random Forest

	feature	importance
0	admission_type_id	0.000942
1	readmitted	0.988307
2	gender	0.000265
3	age	0.001016
4	diabetesMed	0.000827
5	race	0.000658
6	num_medications	0.002725
7	number_diagnoses	0.003896
8	time_in_hospital	0.001364

Increases model accuracy but decreases the interpretability of the model

Out of Bag Classification Accuracy:

100% accuracy in the model

THANK YOU