Machine Learning Fundamentals Capstone Project

Author: Jan Mandinec

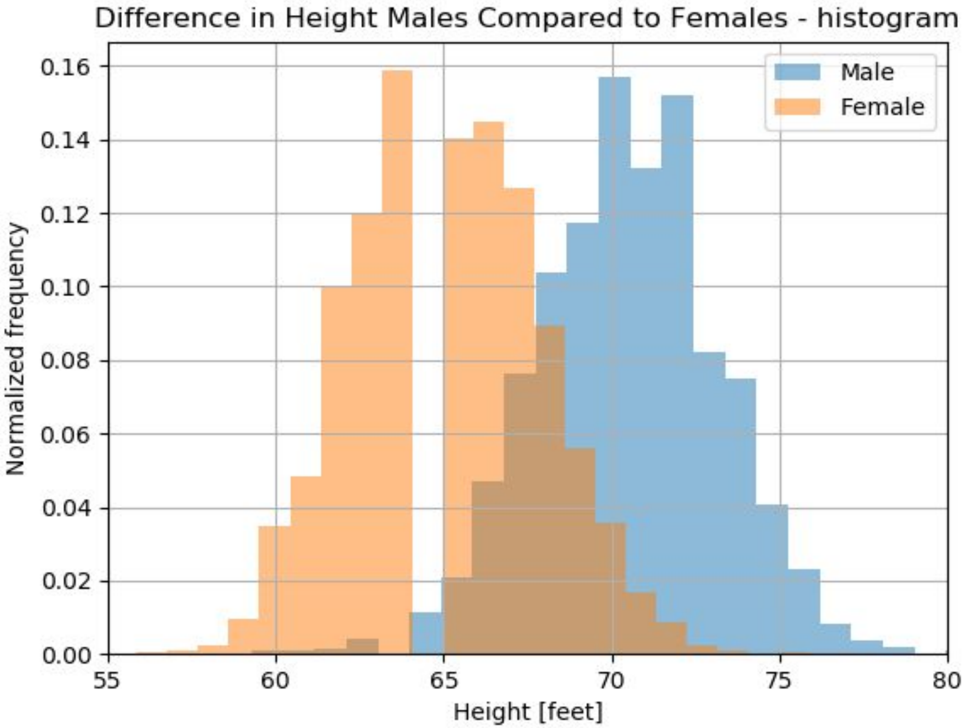
Date: 12/10/2018

Table of Content

1. Exploration of the Data Set
2. Question Statement
3. Augment Data
4. Classification models
   a. Question 1
   b. Question 2
5. Regression models
   a. Question 3
   b. Question 4
6. Conclusion

Exploration of the data set

Men vs Women: Height Difference



Graph shows differences in height between men and women

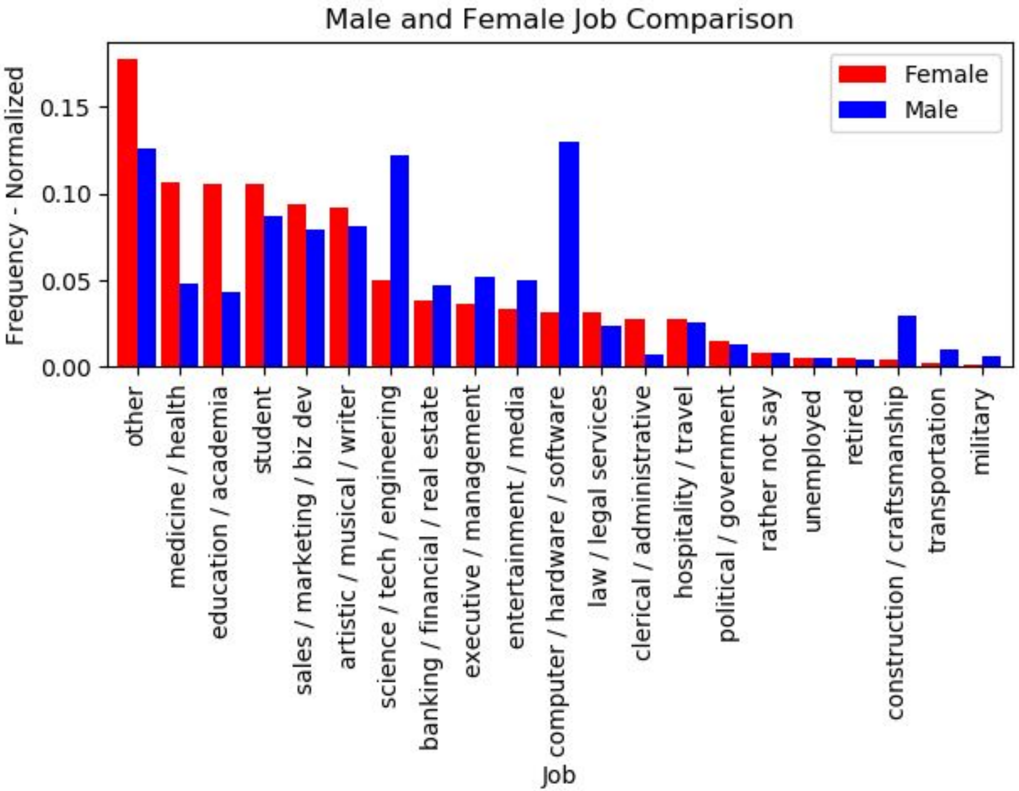
How it was created:

- From main data frame two different data frames for both men and women were created.
- From each frame histograms showing height distribution was created.

Insight:

- Men are generally higher than women

Men vs Women: job comparison



Graph shows what is the division between men and women in workplaces.

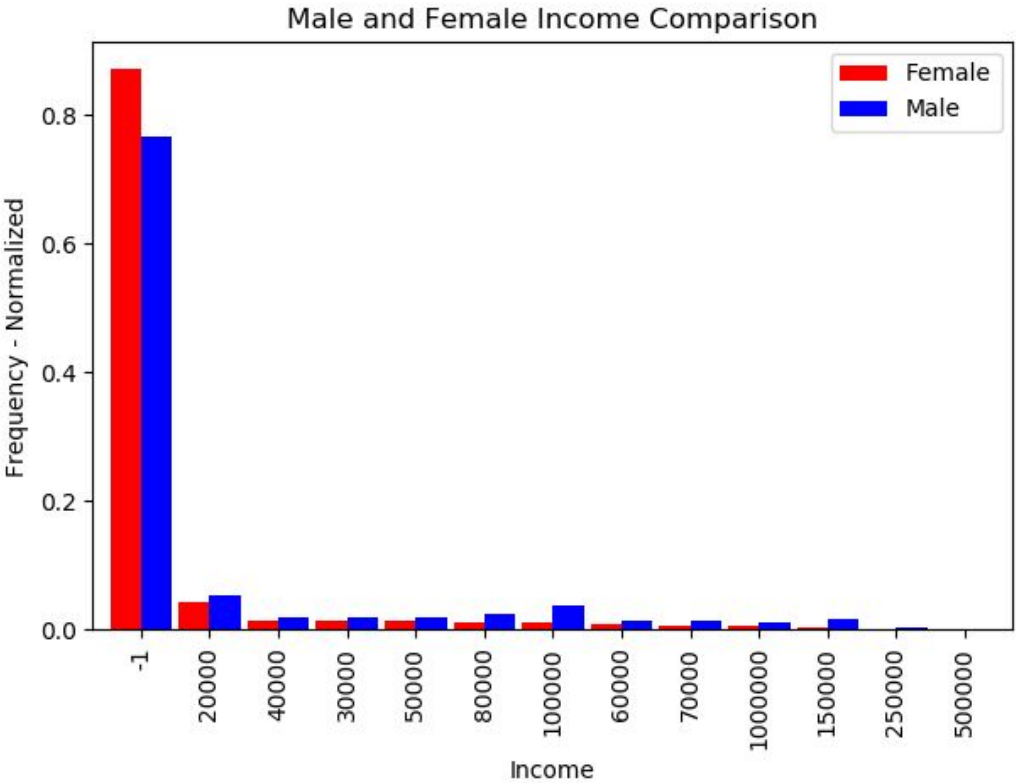
How it was created:

- Data for men and women were counted separately via pandas value_counts
- Data for each group was normalized separately as well.

Insight:

- At some jobs number of men are much higher than number of women (computer science etc.)

Men vs Women:income comparison



Graph shows income difference between men and women.

How it was created:

- Graph was created in exact same way as previous graph.

Insight:

- Among people who doesn’t have any income women dominate.
- Men generally earn more than women.

Questions statements

Questions

As seen from the figures on previous slides, differences between men and women are significant.

Question for Classification:

Question 1: “Can be sex determined just by income and height?”

Question 2: “How will the predictive power of models improve, if job and body_type variables are introduced?”

Question for Regression:

Question 3: “Can be sex predicted only by height with linear regression?”

Question 4: “Will score of the model improve if additional variables are introduced?”

Augment data

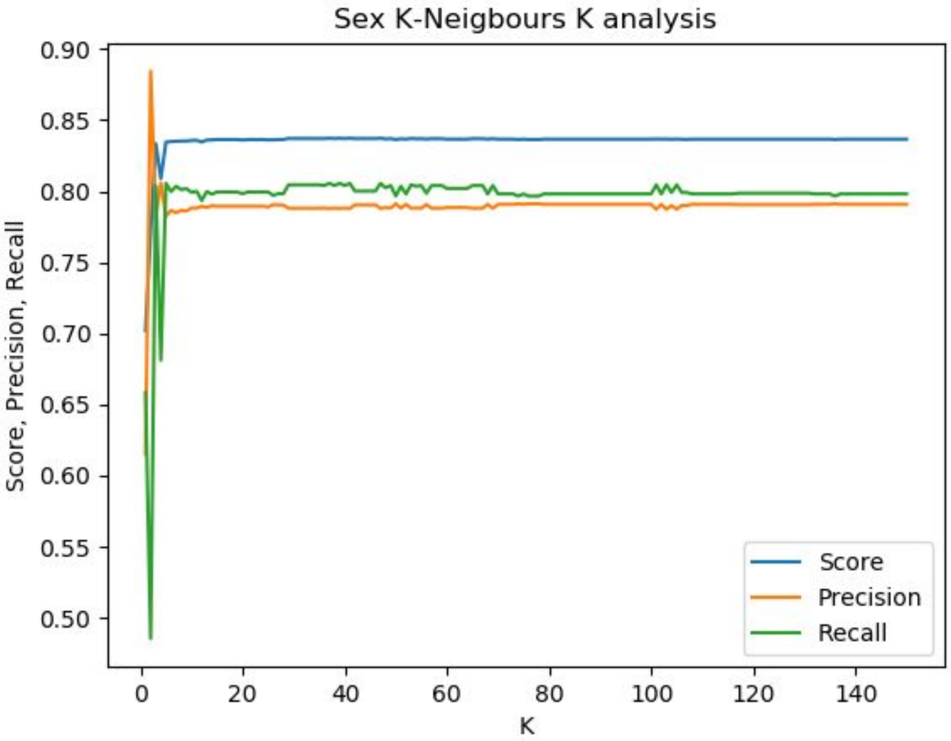
Data for all questions were processed with following procedure:

1. Labeled data was transformed to the code (eg. sex column was transformed: ‘m’ -> 0, ‘f’ -> 1)
2. New columns were added to frame (eg. job_code (see slide 17), body_type_code (see slide 17) etc.
3. All examined data was put into separate dataframe
4. All NaN in examined columns were deleted with whole rows.
5. All data (only classification) was normalized by MinMaxScaler function from sklearn
6. Data was split into training group and test group with train_test_split function from sklearn

Classification models

Question 1: “Can sex be determined just by income and height?”

K-Neighbors model



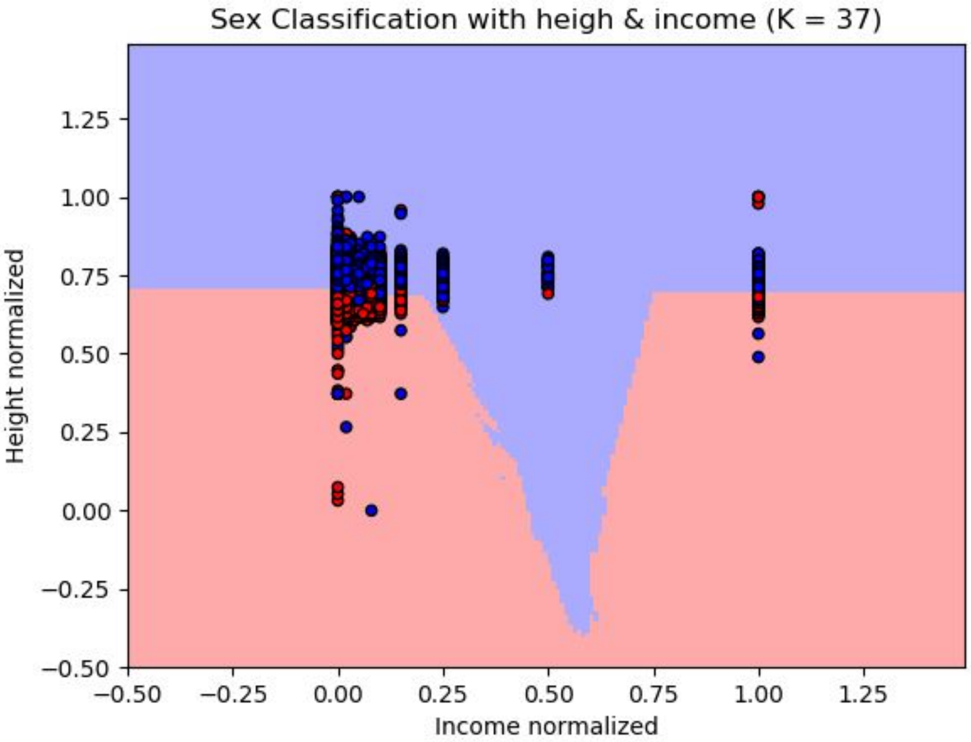
How it was created (figure shows score, precision & recall with respect to changing K value)**:**

- For loop was introduced with n_neighbors = K, as K changes from the range 1 to 150
- For every K KNeighborsClassifier from sklearn was used
- Score etc. for each iteration was stored.
- Best score indicates proper K value

Insight for K-Neighbors K Analysis

- K = 37 provide best score (83.71 %)
- Because of sample size many n_neighbors are needed before overfitting occur.
- Time to compute: 760.11 s

K-Neighbors model



How it was created Figure shows decision boundary for K-Neighbors model for K = 37:

- Points were plotted (income/height for each sex)
- KNeighborsClassifier was used to label (color) mesh (weight = ‘uniform’)
- Time to compute K-Neighbors = 2.59 s
- Numerical results can be seen on slide 15.

Insight: As seen, the blue boundary (males) dominates at the center of the figure. It is probably due to the fact that in the central part of the income spectrum there are significantly more men than women.

Support vector Machines

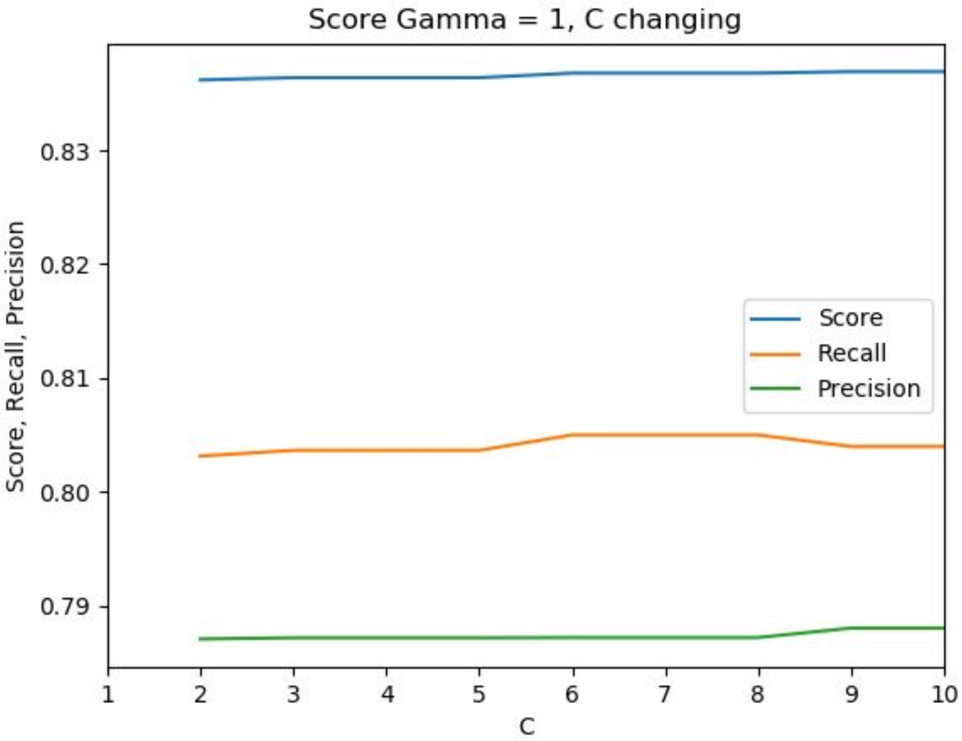


Figure shows distribution of score, recall and precision for Gamma = 1 and C changing

How it was created:

- Nested for loop (for loop in the for loop) was introduced.
- Gamma changes in range from 1 to 5
- C changes in range from 1 to 10
- SVM function from sklearn was performed
- Values (score etc.) for every iteration were saved.
- Time to compute: 3184.06 s

Insight:

- There are not much difference between the results. Best gamma = 1, C = 8

Support vector Machines

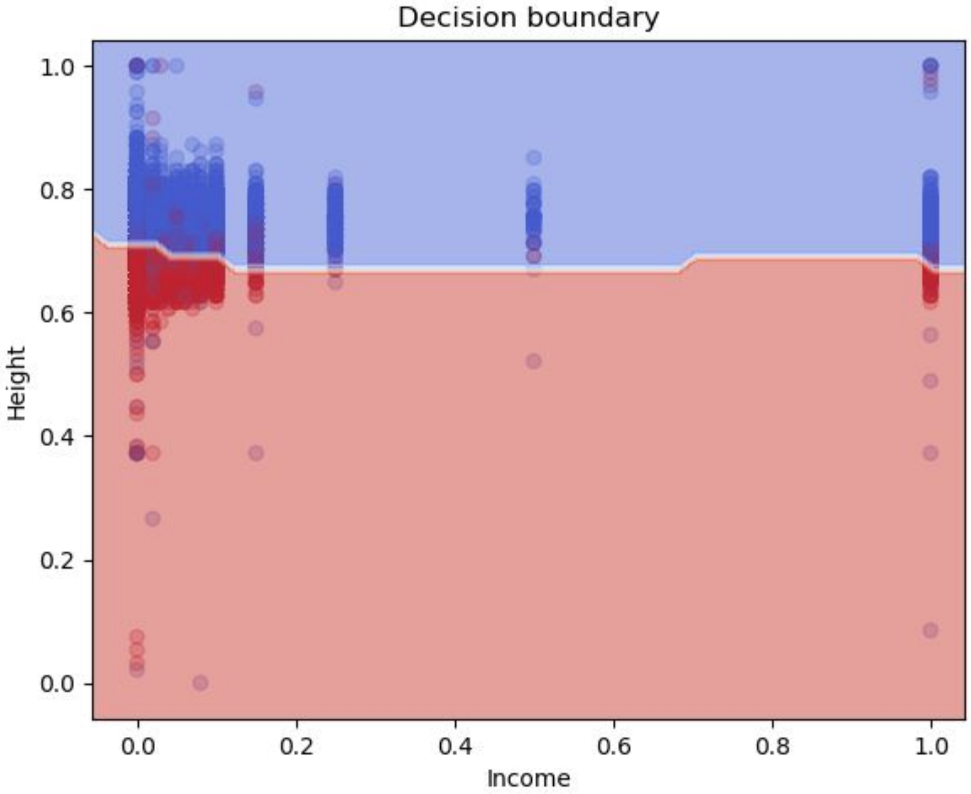


Figure shows scatter plot of height and income with decision boundary from SVM model.

How was created:

- Blue dots = men, Red dots = women
- SMV was calculated for gamma = 1, C = 8
- SMV used ‘rbf’ kernel
- Function draw_boundary from course was used to draw a boundary line
- Numerical results can be seen on slide 15.

Insight: Decision boundary is grasped better for SVM than K-Neighbors.

Question 1: Comparison

Time to find optimal K, respectively gamma & C

	K-Neighbors	SVM
Range K	1 : 150	-
Range Gamma	-	1 : 5
Range C	-	1 : 10
Time average: one iteration	5 s	63 s
Time: whole calculation	760.11 s	3184.06 s

Results for Question 1

	K-Neighbors	SVM
	K = 37	gamma = 1 , C = 8
Score	83.72%	83.69%
Recall	80.55%	80.40%
Precision	78.78%	78.80%
Time to compute	2.59 s	62.86 s

Finding proper parameters:

- Time to calculate parameters for SVM was significantly more than for K-Neighbors
- Reason was that data weren’t forming actual clusters (see slide 14)

Results:

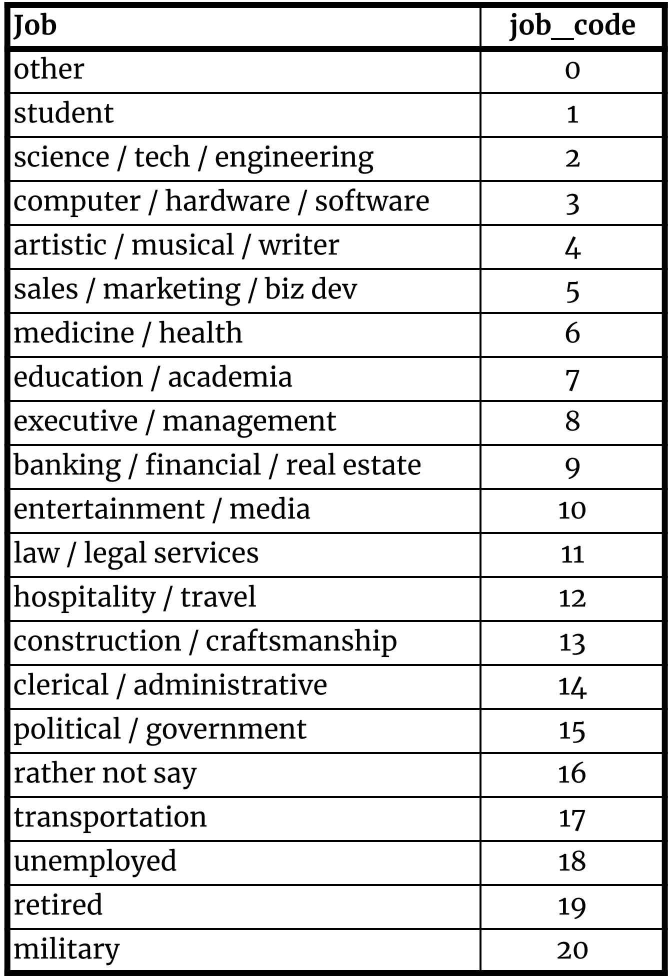
- Similar results (score, recall, precision)
- Main difference - time to compute SVM (‘rgb’) is significantly larger than K-Neighbors (more than 24 times)

Classification models

Question 2: “How will the predictive power of models improve, if job and body_type variables are introduced? ”

Add Features: Job

- Table shows how the job data are transformed to code
- Each job type was described as a number (see table)
- job_code column was added to the data frame
- job_code is used in question 2 to predict sex



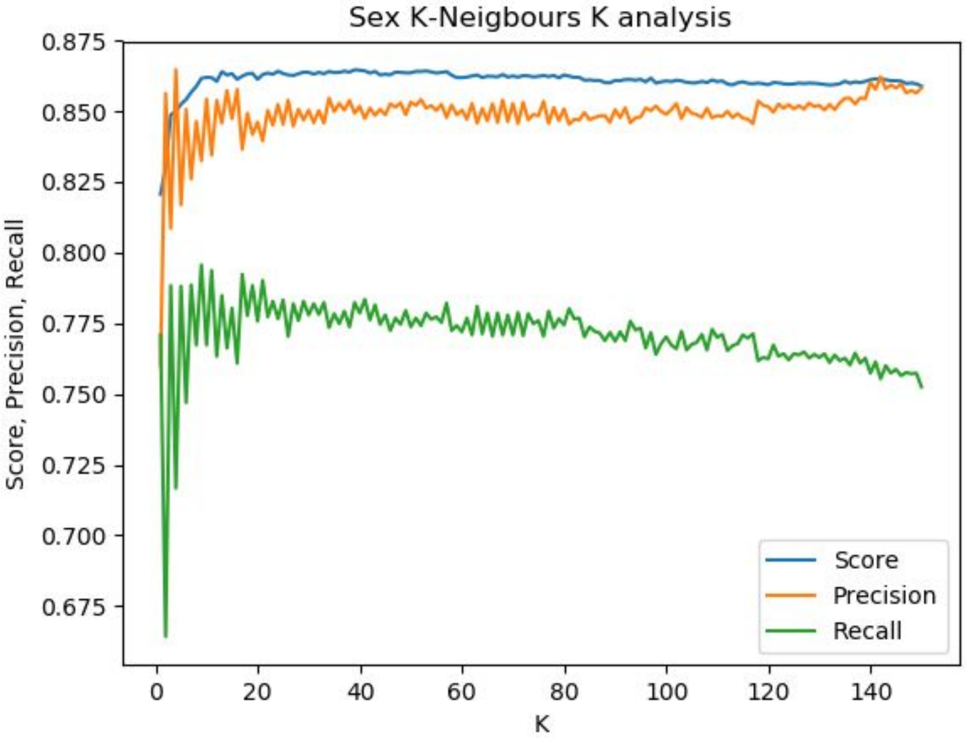
Job	job_code
other	0
student	1
science / tech / engineering	2
computer / hardware / software	3
artistic / musical / writer	4
sales / marketing / biz dev	5
medicine / health	6
education / academia	7
executive / management	8
banking / financial / real estate	9
entertainment / media	10
law / legal services	11
hospitality / travel	12
construction / craftsmanship	13
clerical / administrative	14
political / government	15
rather not say	16
transportation	17
unemployed	18
retired	19
military	20

Add Features: Body_type

- Table shows how the body_type data are transformed to code
- Each body_type was described as number (see table)
- body_code column was added to the data frame
- body_code is used in question 2 to predict sex

body_type	body_type_code
average	0
fit	1
athlete	2
thin	3
curvy	4
a little extra	5
skinny	6
full figured	7
overweight	8
jacked	9
used up	10
rather not to say	11

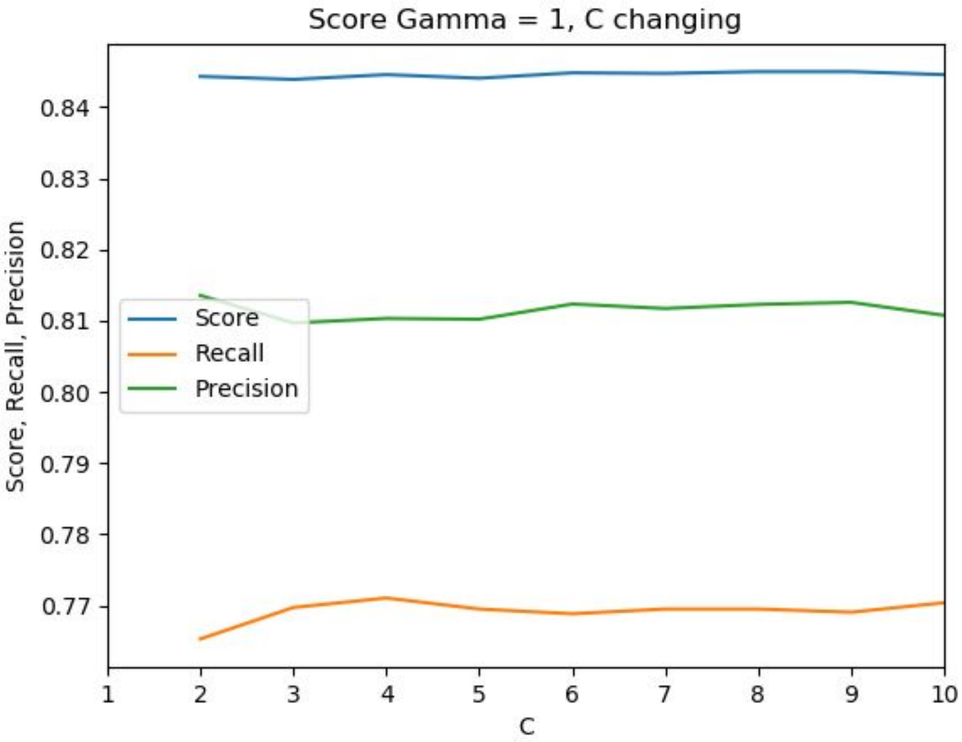
K–Neighbors model



Insight for K-Neighbors:

- Graph shows distribution of Score, Precision and Recall with respect to K. It was produced same way as on slide 11.
- Time to compute proper K value: 488.17 s
- Optimal number of K = 39 (86.46 %)
- As seen from figure, precision is getting slightly bigger as the K value is getting larger. That shows that classifier tends to classify less points with positive labels even though they are actually negative.
- Recall on the other hand becomes smaller, indicating that less relevant (positive) point are selected from the database.

Support vector Machines



As previously gamma and C values were determined firstly. Figure shows the distribution of Score Recall and Precision with fixed gamma value (gamma = 1)

Insights to SVM:

- Graph was produced in the same way as the graph on slide 12.
- As seen from the figure changing C value do not change results much.
- Best Results were obtained with gamma = 5 and C = 10
- Time to compute = 1680 s.

Question 2: Comparison

K-Neighbors comparison

	Question 1	Question 2	Difference
	K = 37	K = 39	
Score	83.72%	86.46%	2.74%
Recall	80.55%	78.23%	-2.32%
Precision	78.78%	84.92%	6.12%
Time to compute	2.59 s	1.43 s	- 44,79%

SVM Comparison

	Question 1	Question 2	Difference
	gamma = 1, C = 8	gamma = 5, C = 10	
Score	83.69%	86.57%	2.77%
Recall	80.40%	77.95%	-3.04%
Precision	78.80%	80.75%	2.41%
Time to compute	62.86 s	29.03 s	- 53,81%

Apart from results from Question 1 in Question 2 two features (body_code and job_code) were added to the data set (see slides 17 and 18).

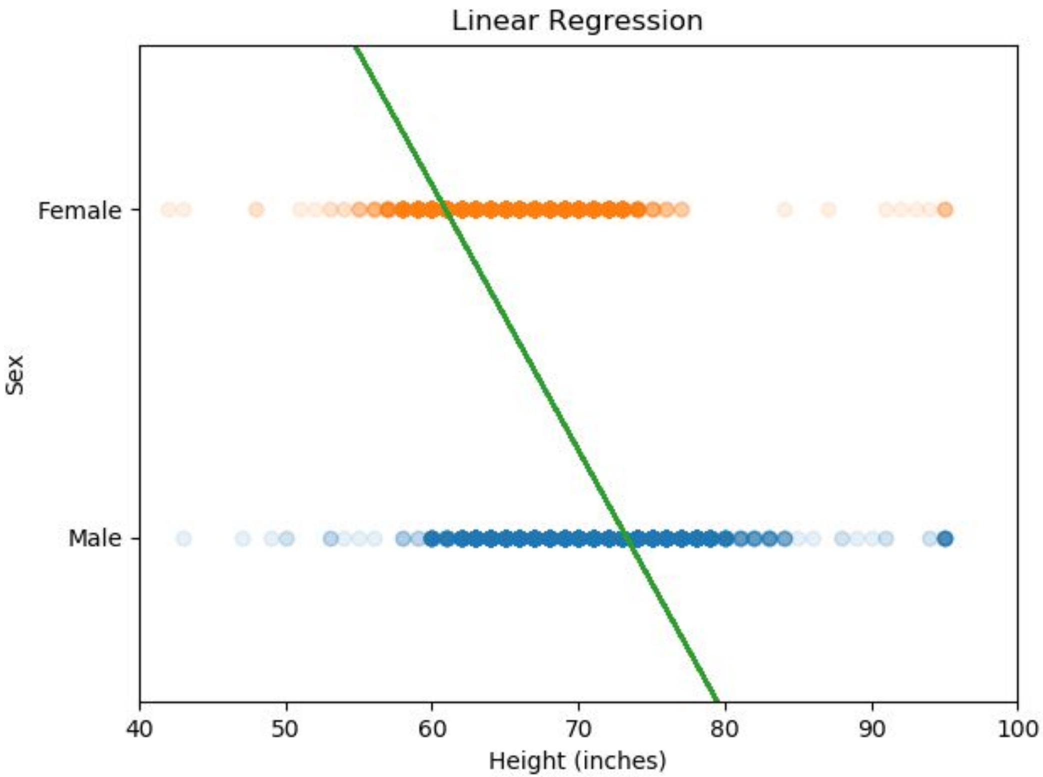
Insights: By introducing new features, models in both cases improved. Improvements are however just in few percentages.

Interesting finding was that by introducing more features to the model time to compute was actually reduced. This reduction was also very significant as the time to compute drop was in both cases more than 40%.

Regression models

Question 3: “Can be sex predicted only by height with linear regression?”

Linear Regression

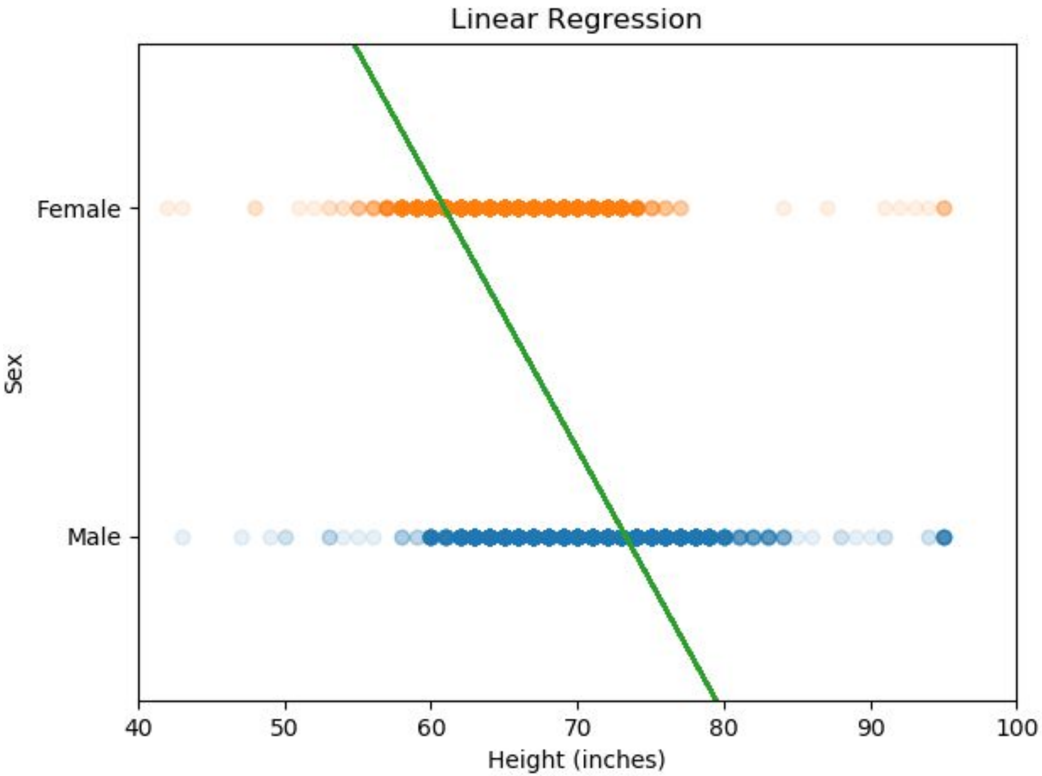


How was created (figure shows linear regression between the heights of males and females)**:**

- LinearRegression model from sklearn was used.
- All males and females were plot with respect to height
- Predicted values from regression model was plotted

Insights: Compare to the classification approach linear regression has proven to be much faster. It took only 0.009 s to calculate. However, the score of the model, as it can be also seen from the figure, suffered.

Linear Regression



Results

- Slope: - 0.08068
- Intercept: 5.9146
- Train score: 43.12%
- Test score: 42.30%
- Time to compute: 0.009 s

Results can be seen on slide 27 where the results are compared to multiple linear regression model.

Regression models

Question 4: “Will score of the model improve if additional variables are introduced??”

Multiple Linear Regression

Features: For multiple linear regression several features were added. Regression was solved with:

- Height as **height** (base from linear regression, given from the start)
- Income as **income (given from the start)**
- Body_type as **body_code** (see slide 18)
- Counted number of words from essays that are significant for men and women as **woman _words_count** & **man_word_count** (see below)

Female & male words from essays:

- NaN in essays were deleted & essays were merged together
- Number of occurrences were count in the essays were counted and store in new series.
- Some of the most frequent words fo men and women were taken from (https://bit.ly/1sEVc1G)

Occurences of word for men and women

Women words	["love", "excited", "happy", "chocolate", "cute"]
Men words	["sport", "shit", "guy", "fifa", "ps3"]

Multiple Linear Regression

Multiple linear regression:

- Multiple Linear Regression was calculated with sklearn model LinearRegression (‘rgb’)
- Results can be seen from the table.
- Results are compared to linear regression model from question 3.

Comparison of results for question 3 and 4

	Linear regression	Multiple linear regression	Difference
Train score	43.12%	46.59%	3.47%
Test Score	42.30%	47.47%	5.17%
Slope	-0.08068	*	-
Intercept	5.91459	5.57282	-5.77%
Time to compute	0.01 s	0.0283 s	64.66%

*** Slope multiple linear regression:**

heigh:	-7.6885e-02
body_code:	2.2051e-02
Income:	-1.6891e-07
woman_words_count:	2.3008e-02
man_words_count:	-4.2517e-02

Results: As seen from the table, result with using multiple linear regression slightly improved as difference in test score was 5.17% and train score was 3.47% in favor of multiple regression. Time to compute also increase significantly, but result is still less than 0.1 s.

Conclusion

Conclusion: Question 1

Question 1: “Can be sex determined just by income and height?”

Both supervised machine learning models (K-Neighbor, Support Vector Machine) showed good results as they were able to predict correct labels with success rate around 83%.

As more convenient model to answer this question, was shown to be K-Neighbors as computational time was significantly lower.

Conclusion: Question 2

Question 2:“How will the predictive power of models improve, if job and body_type variables are introduced?”

By introducing new features (job, body_type), models showed only minor improvement (around 3%). However, huge difference was achieved in computational time as the models with more features showed to be actually faster by not negligible margin (up to 53%). In this case, it is therefore advantageous to add more feature to the classification model. Although computational time drop more for SVM approach, K-Neighbors proven to be better, as it still compute much faster. Of course choosing linear kernel instead of rbf kernel would also lead to reduction of computational time. This possibility have not been explored though. Reason for the slowness of the SMV is probably the fact that data does not form nice clusters. Proper boundary line is therefore harder to find.

Conclusion: Question 3 + 4

Question 3: “Can be sex predicted only by height with linear regression?”

Linear regression didn’t perform very well in this case. Model showed only ordinary success as it was capable of predicting sex in roughly 43%. This number is slightly better than random pick from the data set, where men dominate, as there are 35829 of men (59.77%) and 24117 of women (40.23%). Linear regression was very fast as it calculate the problem in 0.01 s.

Question 4: “Will score of the model improve if additional variables are introduced?”

By introducing other features (body_type, income, woman_words_count, men_words_count), regression model improved slightly as it was able to predict sex in approximately 47,5% of the time. This number is still not very impressing but it proves that multiple linear regression (namely new features) helps to improve model score. Linear regression in general can be considered as not very suitable for this kind of questions.

Conclusion: data wish + future work

Question from CodeCademy: “What other data you would like to have in order to better answer your question(s)”:

As main question of the project was to distinguished sex. I would like to have data where differences between sexes should be visible for a first sight. Data like sports or hobbies would be highly appreciated.

Future work:

It is clear, that in my project I scratched only a surface of the possibilities that these data have. I had troubles to find data where the regression would provide a nice fit. Therefore, I would like to explore some regression possibilities on this data a little further. In my opinion, essays provide another huge opportunities that haven’t been explored much in this project. For instance, from essays one can estimate what kind of persons the individuals really are. Ultimately, I would like to use machine learning techniques to find a match for all people on the data set.