INNOPOLIS UNIVERSITY, DEPARTMENT OF COMPUTER SCIENCE

Theoretical homework #1

Kamil Salakhiev

April 12, 2016

1 BEST MODEL DESCRIPTION

1.1 DATA PREPARATION

Data was written using pandas library:

```
train = pd.read_csv("train.csv", sep=',', header = 0)
```

To extract features and categories appropriate methods get_features() and get_categories were prepared:

```
        X = get_features(train)
        y = get_categories(train)
```

```
def get_features(df,  encoder=None):
        cat=['workclass','education','marital-status','occupation', \
                'relationship', 'race', 'sex', 'native-country']
        num=['age','capital-gain', 'capital-loss', 'hours-per-week']
        if (len(cat) == 0 or len(num) == 0):
                return 'con'
        if encoder is None:
                Xseries_categorical = [get_one_hot_encoding(df[x]) \
                 for x in cat]
        else:
                Xseries_categorical = [get_one_hot_encoding_with \
                        (df[x],encoder) for x in cat]
        Xseries_numerical = [df[x] for x in num]
```

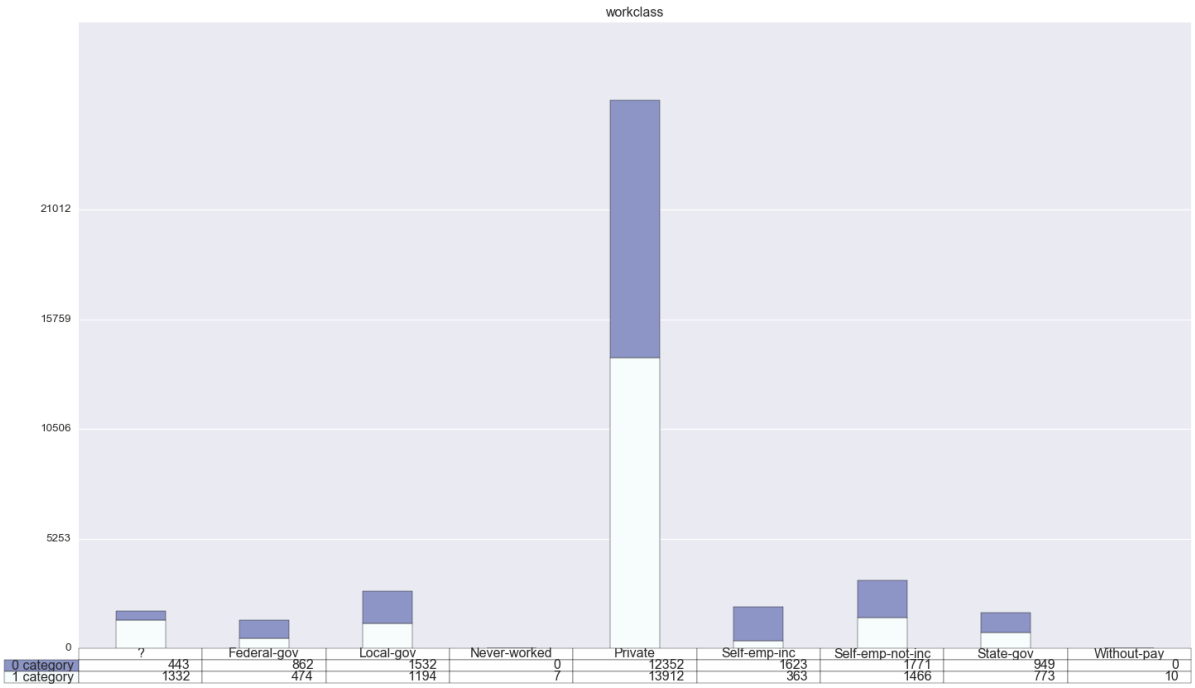
```
        X = pd.concat(Xseries_categorical+Xseries_numerical, axis=1)
        return X.values
def get_categories(df):
        return df['Category'].values
```

To select features, lists cat and num were varied in order obtain the best set. However, eventually, sklearn library were charged for feature selection. Recursive feature elimination were used for this goal, which recursively considered smaller and smaller sets of features. The quality of features' sets were checked by AdaBoost estimator with Decision tree with maximum depth equal one. Before AdaBoost estimator were chosen, plenty of other classificators were considered(see later in report). But AdaBoost with recursive feature elimination gave the best performance. To end up with number of estimators = 900, and SAMME.R algorithm GridSearchCV were used. Calculated cross validation score was 0.870

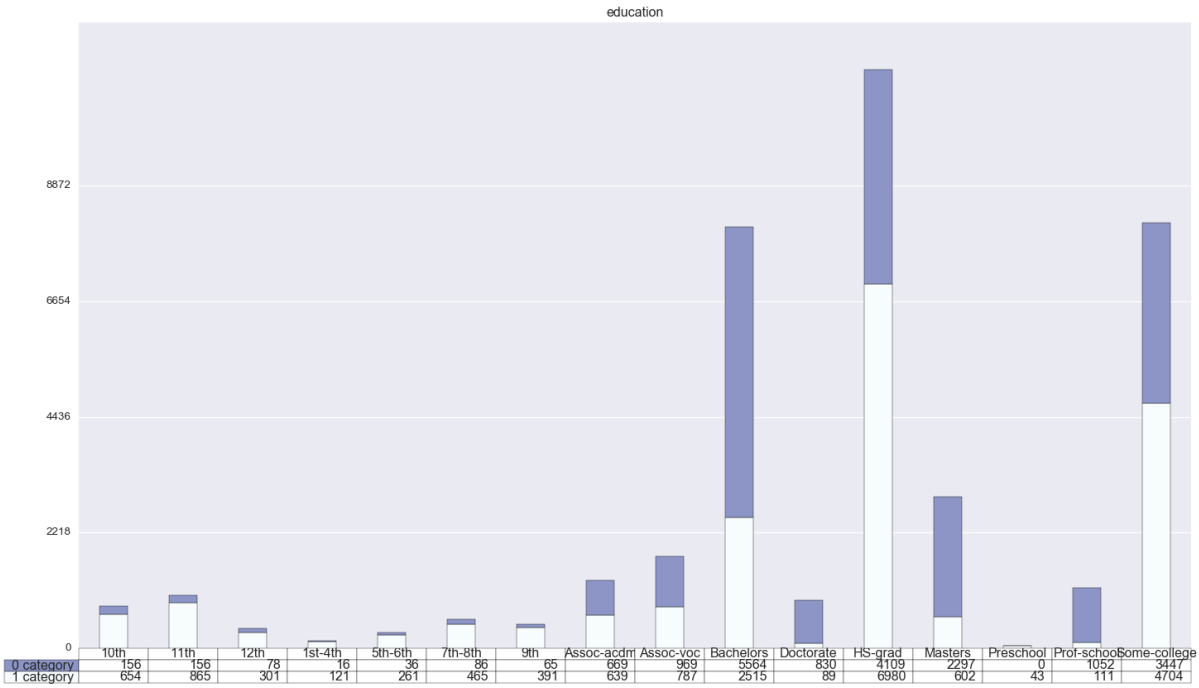
```
ada=AdaBoostClassifier(
        base_estimator=DecisionTreeClassifier(max_depth=1),
        algorithm='SAMME.R',n_estimators=900)
rfecv = RFECV(estimator=ada, step=1, scoring='accuracy',verbose=1)
```

2 GRAPHS

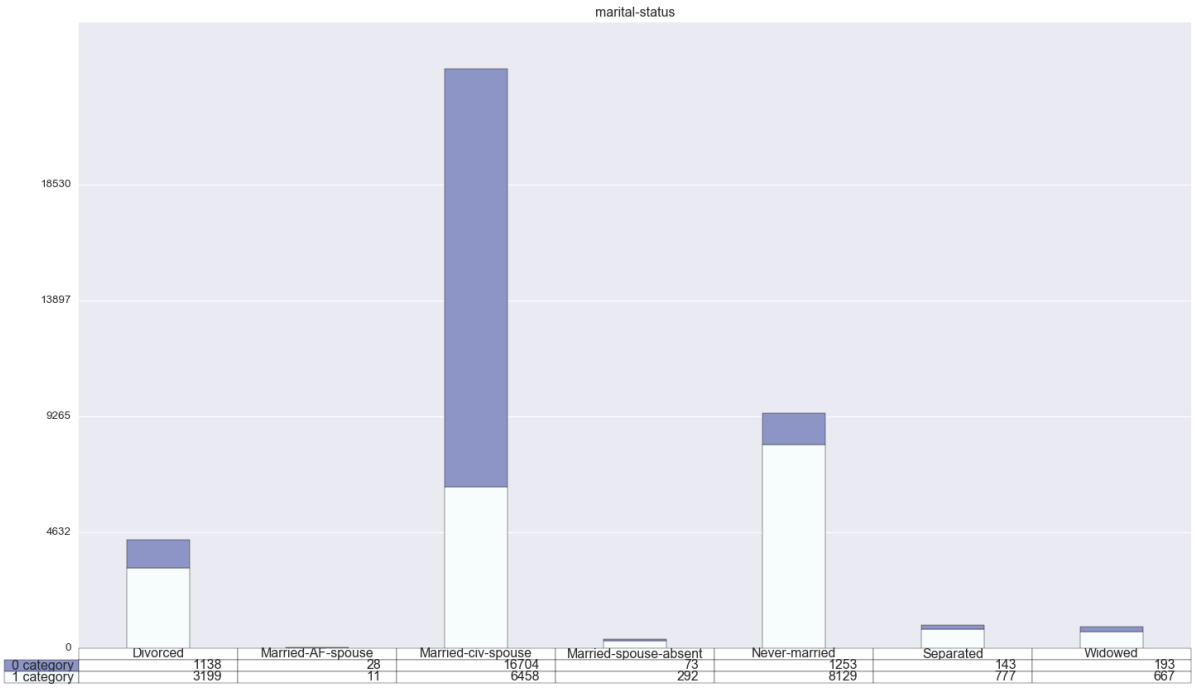
Before I ended up with AdaBoost classifier, many other classifiers were used. To select appropriate features, dataset were visualized. All used graphics are placed below.



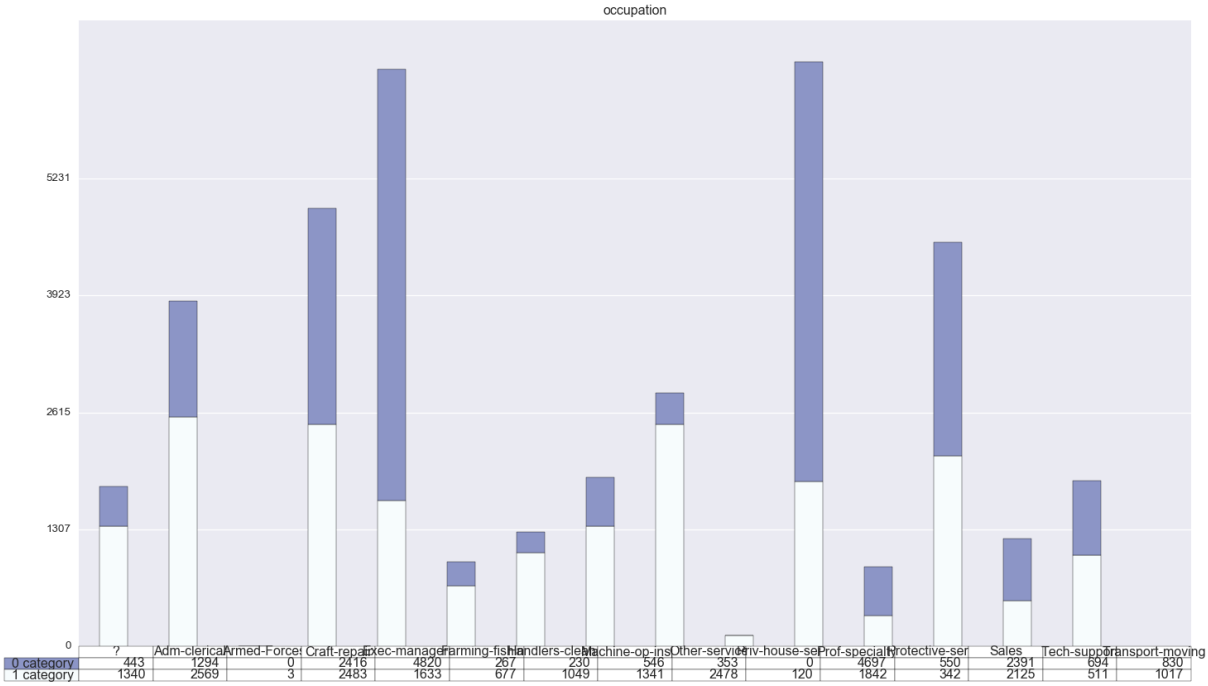
	?	Federal-gov	Local-gov	Never-worked	Private	Self-emp-inc	Self-emp-not-inc	State-gov	Without-pay
0 category	443	862	1532	0	12352	1623	1771	949	0
1 category	1332	474	1194	7	13912	363	1466	773	10



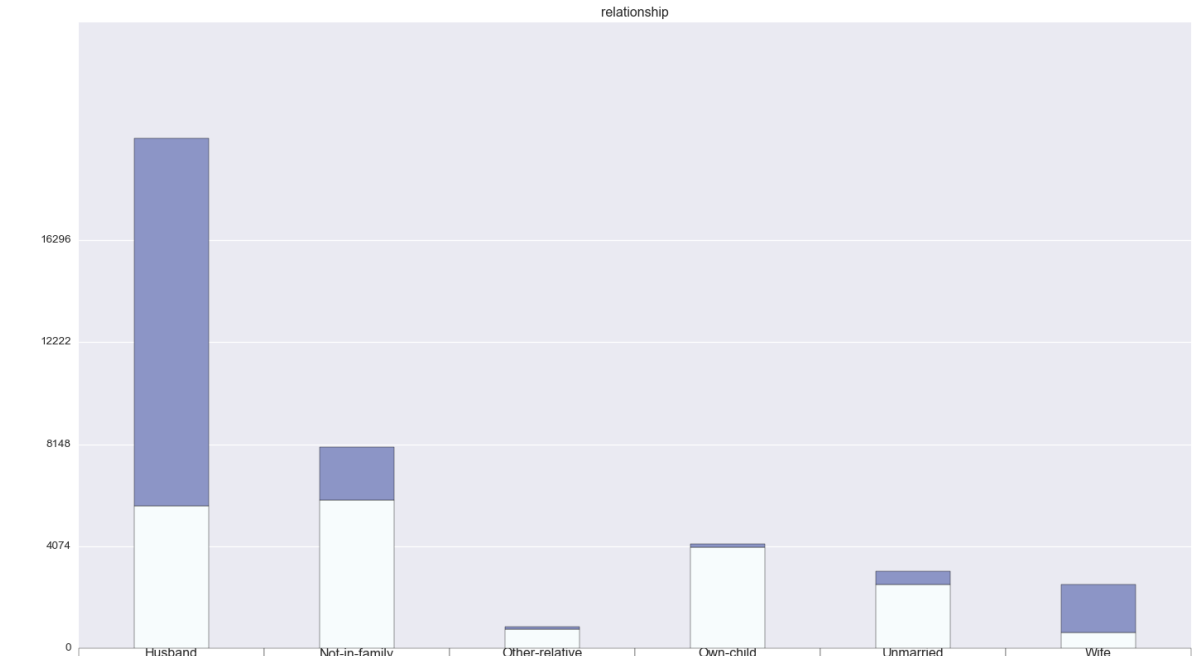
	10th	11th	12th	1st-4th	5th-6th	7th-8th	9th	Assoc-acdm	Assoc-voc	Bachelors	Doctorate	HS-grad	Masters	Preschool	Prof-school	Some-college
0 category	156	156	78	16	36	86	65	669	969	5564	830	4109	2297	0	1052	3447
1 category	654	865	301	121	261	465	391	639	787	2515	89	6980	602	43	111	4704



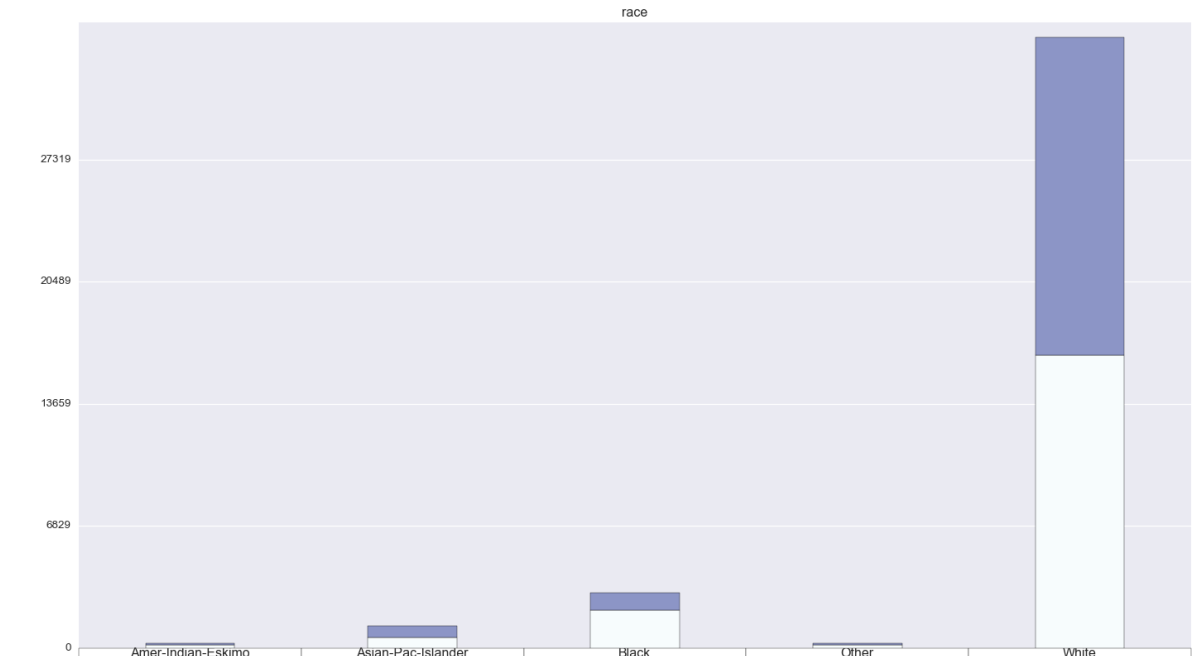
	Divorced	Married-AF-spouse	Married-civ-spouse	Married-spouse-absent	Never-married	Separated	Widowed
0 category	1138	28	16704	73	1253	143	193
1 category	3199	11	6458	292	8129	777	667



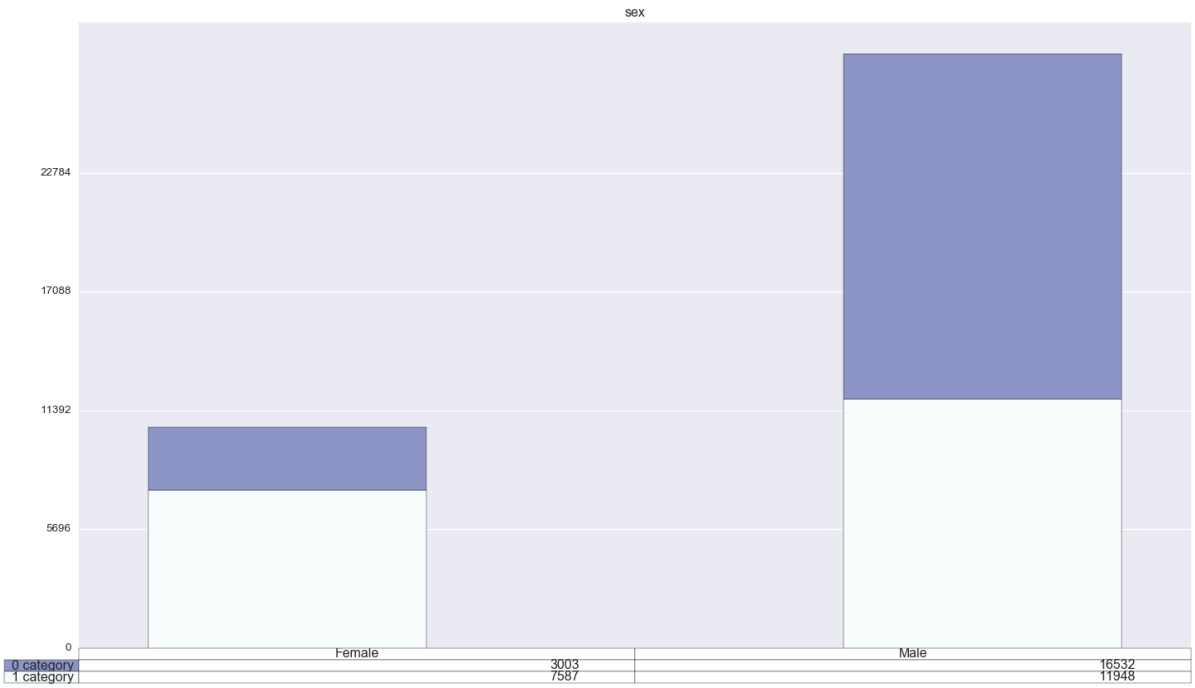
	?	Adm-clerical	Armed-Force	Craft-repair	Exec-manager	Farming-fishin	Handlers-clean	Machine-op-ins	Other-service	Priv-house-se	Prof-specialt	Protective-ser	Sales	Tech-support	Transport-moving
0 category	443	1294	0	2416	4820	267	230	546	353	0	4697	550	2391	694	830
1 category	1340	2569	3	2483	1633	677	1049	1341	2478	120	1842	342	2125	511	1017

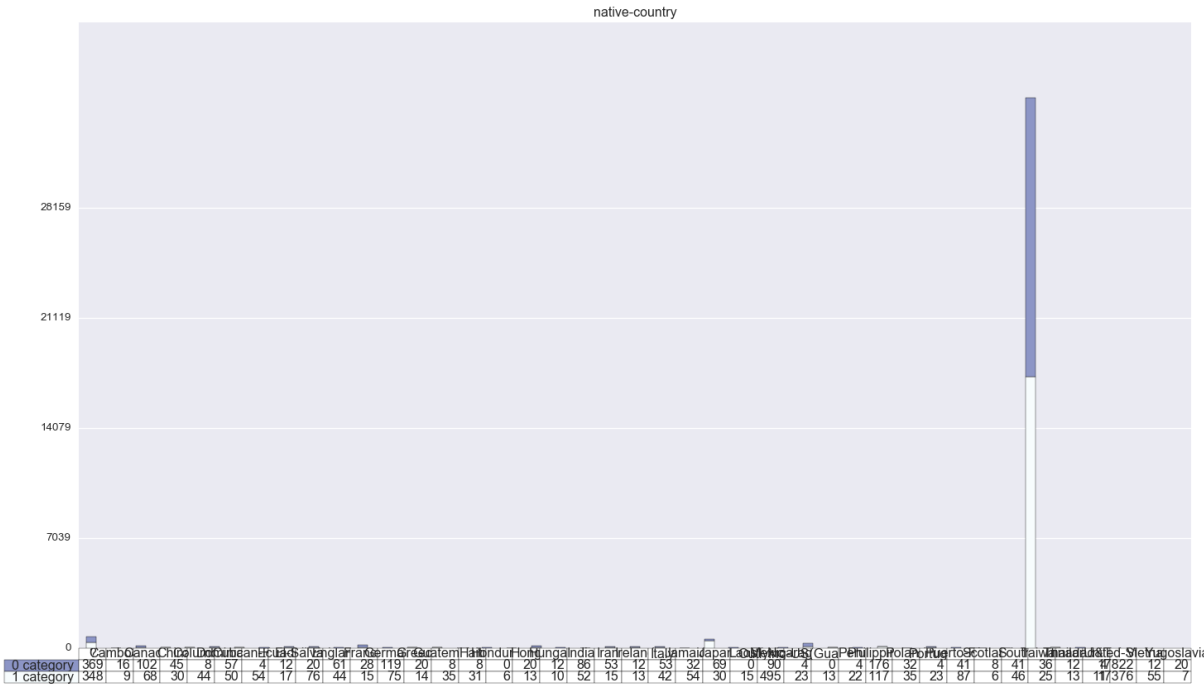


	Husband	Not-in-family	Other-relative	Own-child	Unmarried	Wife
0 category	14695	2128	106	143	546	1914
1 category	5674	5904	750	4028	2542	634



	Amer-Indian-Eskimo	Asian-Pac-Islander	Black	Other	White
0 category	82	661	978	73	17740
1 category	203	607	2125	190	16409


sex
22784
17088
11392
5696
0
Female
Male
0 category
1 category
3003
7587
16532
11948



Based on graphics, I decided to remain the following set of features: ['sex', 'education', 'native-country', 'marital-status', 'relationship', 'occupation', 'capital-gain', 'capital-loss']. Other features showed less certainty. For example workclass feature shows, that based on this feature it is hard to decide to which category person belongs, while marital status demonstrates good reliability. For example if we have 'never-maried' person we can see that this person most likely belongs to the first category.

3 Other models

The following models were obtained with gridsearch (aforementioned features were supplied to the models):

- SVM with RBF kernel and gamma=0.1 with PCA dimensionality reduction to the 40 components. Obtained accuracy:0.862
- Decision tree with PCA dimensionality reduction to 2 components. Obtained accuracy: 0.862
- Random forest classifier with PCA dimensionality reduction to 3 components with max depth 11 and with 15 estimators. Obtained accuracy: 0.863

- KNN classifier with PCA dimensionality reduction to 20 components with 13 nearest neighbors and power distance parameter p = 1. Obtained accuracy: 0.860
- Bagging with KNN with 5 nearest neighbors and p=1. Obtained accuracy: 0.863

After deriving several strong classifiers, they were put into 1 voting classifier. As result I got improving to 0.866 accuracy.
However, again, AdaBoost beat that result with accuracy on cross validation 0.870. Latter model were used for make submission on kaggle with accuracy 0.86809.

Classificator	Accuracy on CV
SVM	0.862
Decision tree	0.862
Random forest	0.863
KNN	0.860
Bagging on KNN	0.863
Voting Classifier	0.866
AdaBoost	0.870

Table 1: Classifers' accuracy