Marriage Status Classification and Clustering

Yong Liang
College of Information and Computer Sciences
University of Massachusetts Amherst, Amherst, MA
yongbinliang@umass.edu

Abstract

In this experiment, we'll use the adults' dataset to predict the marital-status. We will first clean and preprocess the data to work with Scikit-Learn models. We will compare the performance of different models using validation methods and grid-search for optimal hyper-parameters. We first analyze the problem with classification models such as KNN, Decision Trees, Random Forest. Then we'll explore clustering models such as K-means Minibatch, Mean-Shift and DBSCAN. Within the dataset features, several seem to be highly correlated. We can improve performance by using feature selection and dimensionality reduction. Lastly, since clustering is difficult to match 1 to 1 to the target label, we'll compare the distribution of the cluster centroid stats with the true label stats.

1 Introduction

The problem is to classify marriage status given the other features from the dataset. This would be a classification problem. We can also apply it as clustering problem to identify the different groups of adults that are not married.

Hypothesis for marriage status group: Some low income can't afford to have a family so they never married; Some might have too much money and wish to keep the money for self; certain type of jobs' work hours are too stressful that they can't balance their work-life, so they are divorced.

1.1. Data Sets

The dataset I'm using is from "UC Irvine Machine Learning Repository". The data consist of survey data of adult census data with their income information. There are 14 column feature attributes and 50k records with some columns contain empty cells. Titles: Age, workclass, fnlwgt, education, education-num, marital-status, occupation, relationship, race, sex, capital-gain, capital-loss, hours-per-week, native-country, income

1.2 Related Works

The original use of the adults dataset was to predict whether someone will have income exceeds $50K/yr based on features of their census data. The models use in the original research was Naïve Bayes, KNN, SVM and others. The highest performance was around 86% accuracy. Since I will be using the features and labels differently, the performance will not be comparable to the original research.

1.3 Data Cleaning

I will use "marital-status" as my label, and drop feature "fnlwgt" and "relationship". "Fnlwgt" is the estimated weight representation of the census. "Relationship" is a more detail description of marital status. There are about 2500 records with some missing values, I've decided to retain in the model. Since I am using scikit-learn models, I have to convert the features into vector components using LabelEncoding.

1.4 Tools and library

I will be using the numpy and scikit-learn model. Plotting will be done with matlab and excel charts.

2 Model Evaluations

I would establish the benchmark as highest percent marriage status group. Then I would run through different models using cross-validation to evaluate the score performance. To optimize the models, I will use hyper-parameter selection, feature selection, and dimensionality reduction to refine the test accuracy scores.

2.2 Train-Validate-Test

I split the dataset for 36k training set, and 16k for testing set. Within the training set, I held out 20% of the data for model validations. This method of model evaluation is fast compare to other cross validation methods and works well when large dataset that covers most data dimensions have.

2.3 Models Scoring

The supervised learning classification models, I will compare are KNN, Decision Trees, and Ensemble Random Forests. I will compare the score by the number of corrects vs the true labels in the test set.

For unsupervised learning clustering models, I will compare Minibatch K-means. I’ve looked into other methods such as Mean-Shift and DBSCAN that has uneven cluster size, but they were too slow to run on my system. Since the cluster groups does not match to the true label one to one. I will use djusted_mutual_info_score scoring to evaluate the mutual information adjusted for chance between each cluster.

2.4 Hyper-parameter and Feature Selection

I will use grid-search over several parameter spaces for each model with a held-out validation set to find the best combinations.

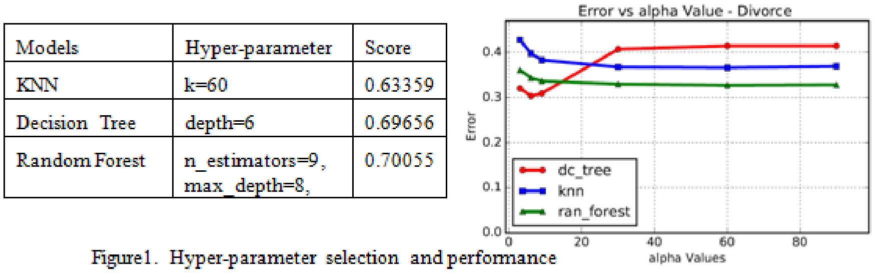
For features selection, I will use K-best, Variance Threshold, LinearSvc, and ExtraTrees. First by fitting the training set to the model, then transform the test features with that model and finally score against the test labels.

For dimensionality reductions, I will use PCA and FastICA, since they are fast and scalable with large datasets.

3. Model Performance Results

Classification benchmark: 46% of the survey records are married status.

From Figure1 with hyper-parameter selection, I found Random Forest has the best performance at 0.70.



Models	Hyper-parameter	Score
KNN	k=60	0.63359
Decision Tree	depth=6	0.69656

Random Forest	n_estimators=9,
	max_depth=8,
		0.70055

Figure1. Hyper-parameter selection and performance

Clustering benchmark: 7 types of marital status

I used the MiniBatch K-Means, which assumes even cluster size, flat geometry, and not too many clusters. The model with cluster size of 4, it scored 0.10494. We can see the elbow effect in Figure2, where the score is maximized at K=4.

I tried Mean-Shift and DBSCAN clustering, which assumes the clusters sizes are uneven and non-flat geometry. However, the models took too long to run.

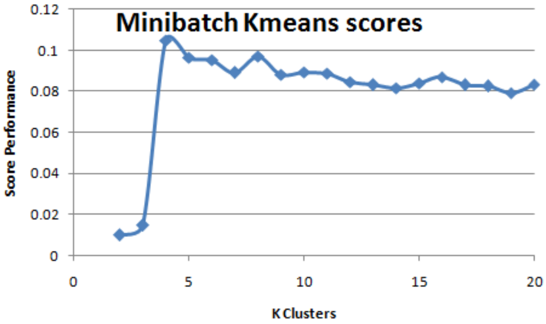


Figure2. Elbow effect on optimal K clusters

4 Performance Improvements:

4.1 Feature Weights

Using ExtraTreeClassifier’s importance feature, we can see the top 3 features highlighted in yellow on the right. “Work-hours” and “income” was an important feature as we see the distributions in the Figure4 and Table2 below. Gender has high unusual weight perhaps due to bias surveying methods.

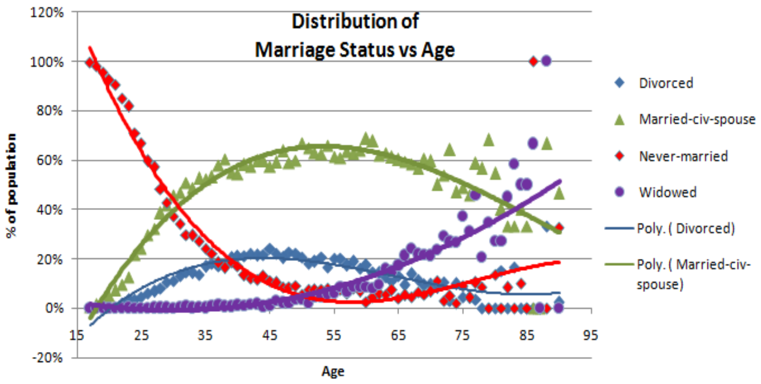
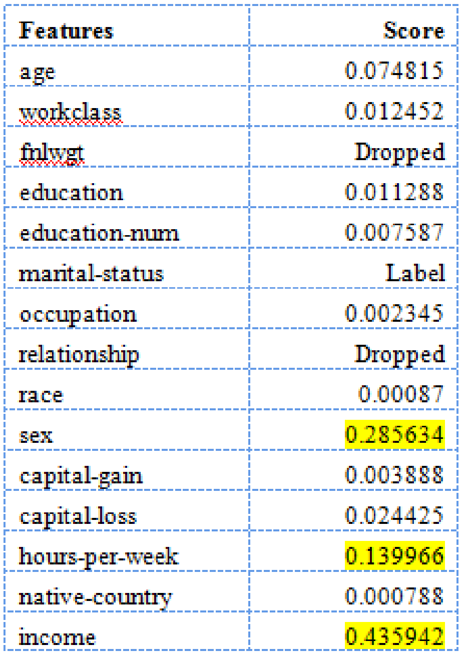


Figure3. Population % by age ranging from 17 to 90+



Features	Score
age	0.074815
workclass	0.012452
fnlwgt	Dropped
education	0.011288
education-num	0.007587
marital-status	Label
occupation	0.002345
relationship	Dropped
race	0.00087
sex	0.285634
capital-gain	0.003888
capital-loss	0.024425
hours-per-week	0.139966
native-country	0.000788
income	0.435942

Tabel1. Features and their importance weight

There is a clear pattern of marriage status and the probability of each group given age.

- Younger people less than 30 are mostly “Never married”, inversely relates to “married”
- As age increase beyond 65, there are more likely people loss their spouse and become “windowed”
- “Divorced” population tends to have peak at age 45.

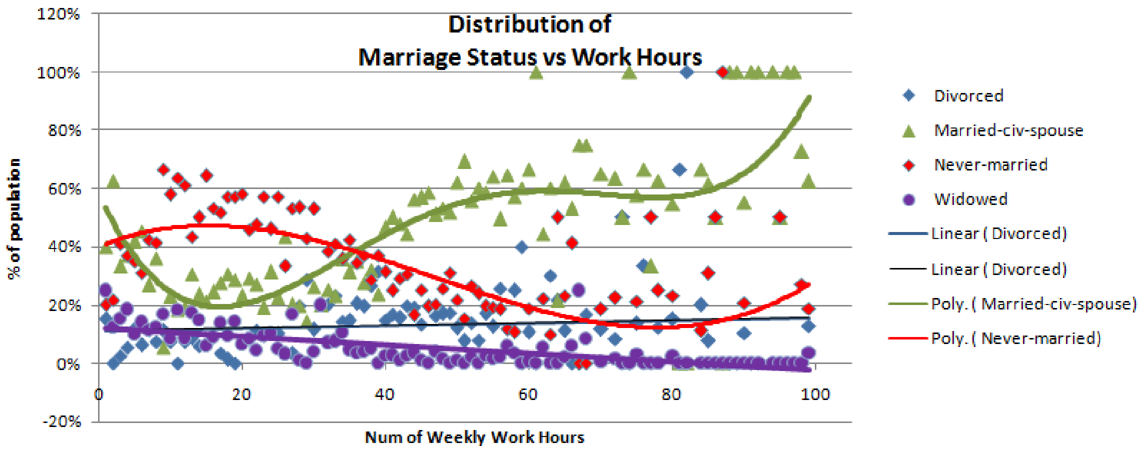
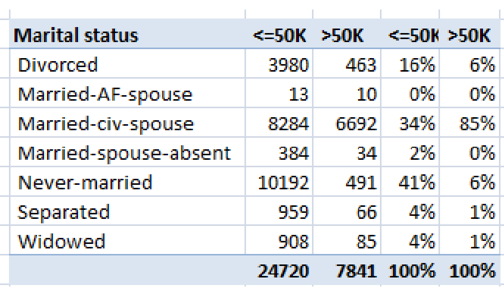


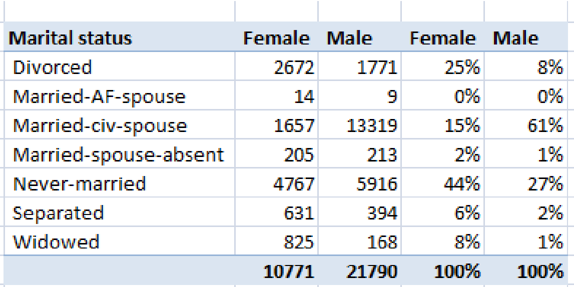
Figure4. Population % by work hours from ranging from 0 to 100+

From this Figure4 work hour distribution chart, below are my analyses on the patterns I found:

- For “never-married”, as work hours increase, people seem to be in a stable job and they start to think about marriage. This looks like it has a high negative correlation with married status.
- For “married”, between 0 and 10 hours part-time, it seems those are stay at home parents that need to take care of children. Longer work hours are usually married couples, perhaps they need to support their children, or in established high demanding positions in their career.
- There is a slight up trend for “divorced” rate as number of work hours increased.
- Most “widowers” have work hours of less than 40, perhaps they are in grief and not interested in working full time.



Marital status	<=50K	>50K	<=50K	>50K
Divorced	3980	463	16%	6%
Married-AF-spouse	13	10	0%	0%
Married-civ-spouse	8284	6692	34%	85%
Married-spouse-absent	384	34	2%	0%
Never-married	10192	491	41%	6%
Separated	959	66	4%	1%
Widowed	908	85	4%	1%
	24720	7841	100%	100%



Marital status	Female	Male	Female	Male
Divorced	2672	1771	25%	8%
Married-AF-spouse	14	9	0%	0%
Married-civ-spouse	1657	13319	15%	61%
Married-spouse-absent	205	213	2%	1%
Never-married	4767	5916	44%	27%
Separated	631	394	6%	2%
Widowed	825	168	8%	1%
	10771	21790	100%	100%

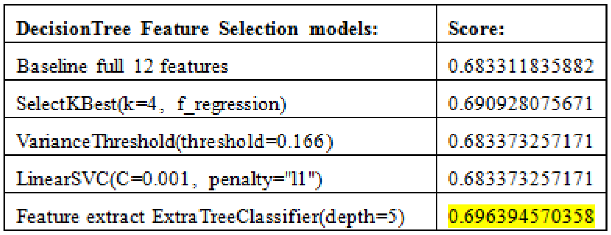
Table2. Marital Status by Income and Marital Status by Gender

From Table 2, we can see there is a clear difference between the groups within the two features.

- Higher incomes tend to be married.
- Lower income tend to be unmarried or divorced
- Females reported higher divorces, separated, windowed and never married
- Males reported mostly as married.

4.1 Classification Feature Selection

From the feature weights in Table 1, we saw many of the features have low weights. With feature selection, we can eliminate the noise in the model and improve the score performance. In Table3 on the right, ExtraTreeClassifer resulted in highest score improvement.

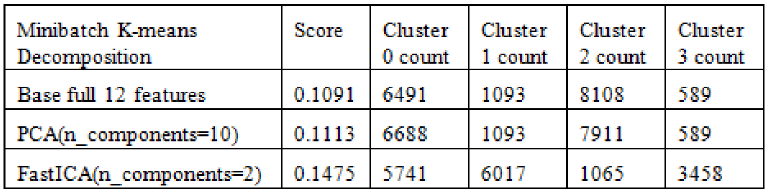


DecisionTree Feature Selection models:	Score:
Baseline full 12 features	0.683311835882
SelectKBest(k=4, f_regression)	0.690928075671
VarianceThreshold(threshold=0.166)	0.683373257171
LinearSVC(C=0.001, penalty="l1")	0.683373257171
Feature extract ExtraTreeClassifier(depth=5)	0.696394570358

Table3. Model improvements by different feature selection methods

4.2 Clustering Dimensionality Decomposition

From Table4, we see that FastICA with 2 components significantly improved the clustering score from the baseline. We don’t know if the clusters labels matchup exactly, so the best way is to compare statistics on the centroids for each cluster.

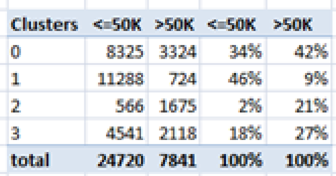


Minibatch K-means Decomposition	Score	Cluster 0 count	Cluster 1 count	Cluster 2 count	Cluster 3 count
Base full 12 features	0.1091	6491	1093	8108	589
PCA(n_components=10)	0.1113	6688	1093	7911	589
FastICA(n_components=2)	0.1475	5741	6017	1065	3458

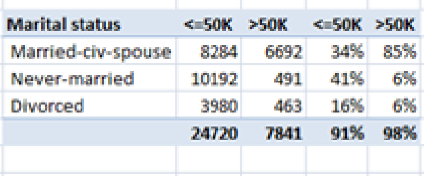
Table4. Score of Minibatch K-means decomposition with 4 clusters, and counts

4.2.1Comparing Clusters Centroids to True Label

From Table5, of FastICA with Minibatch K-Means clusters compare to the actual top 3 marital status class. The distribution has some similarities. For a more in-depth analysis, we can compare feature by feature for similarities.



Clusters	<=50K	>50K	<=50K	>50K
0	8325	3324	34%	42%
1	11288	724	46%	9%
2	566	1675	2%	21%
3	4541	2118	18%	27%
total	24720	7841	100%	100%



Marital status	<=50K	>50K	<=50K	>50K
Married-civ-spouse	8284	6692	34%	85%
Never-married	10192	491	41%	6%
Divorced	3980	463	16%	6%
	24720	7841	91%	98%

Table5. Comparing cluster by income with true label

5. Conclusion and Future Works

Some of my hypothesis about the distribution of the marital status groups was wrong. In fact, most long-hour working people are “married”, lower income people have a higher “divorced” rate, and people working part-time are mostly “never-married”.

For classification models, Random Forest has the best accuracy at 0.70, followed by Decision Tree at 0.683. When using ExtraTreeClassifier feature selection with Decision Tree, the model improved to 0.696

For clustering models, Minibatch K-Means is fast and score of 0.109. Using FastICA decomposition with 2 components, the score improved to 0.148.

One of the challenges I have was computational power. In order to run some of the complex model, I would need to move the model to a compute server. This is hard to do while I am still testing the different model setups. Also grid search over multiple hyper-paramter, feature selection, decomposition would be much slower. Therefore I performed some of these steps sequentially.

There are several issues with the dataset that would be interesting to investigate in the future. For example, there are higher numbers of male than female responded to the survey. This might cause some selection bias in the female group. Another problem I see is the time of the survey, some people might be in multiple groups throughout their life time, or recently switched from married to divorced.

References:

[1] "Adult Data Set." UC Irvine Machine Learning Repository. Center for Machine Learning and Intelligent Systems, n.d. Web. http://archive.ics.uci.edu/ml/datasets/Adult

[2] Bart Hamers and J. A. K Suykens. Coupled Transductive Ensemble Learning of Kernel Models. Journal of Machine Learning Research, Bart De Moor. 2003. ftp://ftp.esat.kuleuven.be/sista/hamers/BH_clm.pdf

[3] Kohavi, Ron, Barry Becker, and Dan Sommerfield. "Improving Simple Bayes." Rexa Paper. Data Mining and Visualization Group Silicon Graphics, Inc., n.d. Web.
http://rexa.info/paper/4c8e8cf6857f1f1bc9b43679d241b096513ee6f2

[4] Rosset, Saharon. "Model Selection via the AUC." (2004): n. pag. IBM T.J. Watson Research Center. Tel Aviv University. Web. http://www.tau.ac.il/~saharon/papers/auc-fixed.pdf

[5] Ron Kohavi, "Scaling Up the Accuracy of Naive-Bayes Classifiers: a Decision-Tree Hybrid", Proceedings of the Second International Conference on Knowledge Discovery and Data Mining, Data Mining and Visualization Silicon Graphics, Inc. 1996 http://robotics.stanford.edu/~ronnyk/nbtree.pdf

[6] "Statistical Consulting Group." Statistical Consulting Group. San Diego State. http://scg.sdsu.edu/dataset-adult_r/