MACHINE LEARNING CS-GY 6923

FINAL PROJECT

RISK PROFILING AND TOLERANCE FOR THE INVESTOR

INSTRUCTOR:

PROJECT PARTNERS

ANSH DESAI (asd9717)
SIDHARTH PUROHIT (sp6365)

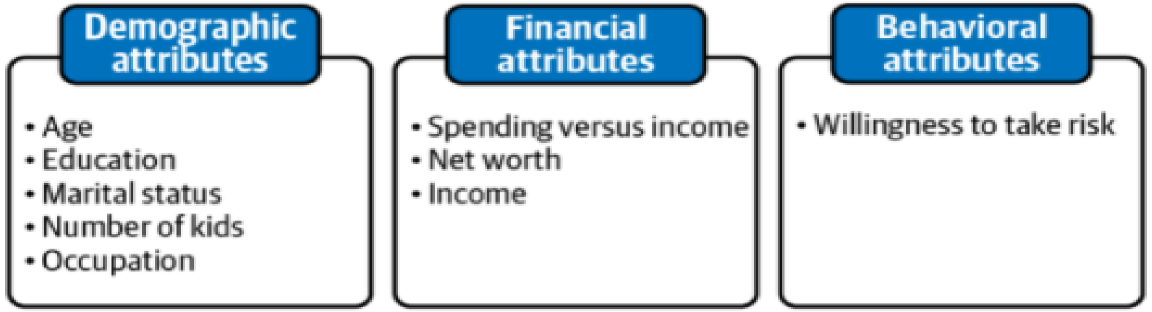
TABLE OF CONTENTS

Name of the topic **Page No.**

Risk Profiling and Tolerance for the Investor

Purpose:

By totally removing the client from the process, we were able to develop a machine learning-based Portfolio management system. The overarching goal is to show how machine learning may be used to automate the manual phases in the portfolio management process. Machine learning is also being used to



Demographic attributes

- Age
- Education
- Marital status
- Number of kids
- Occupation

Financial attributes

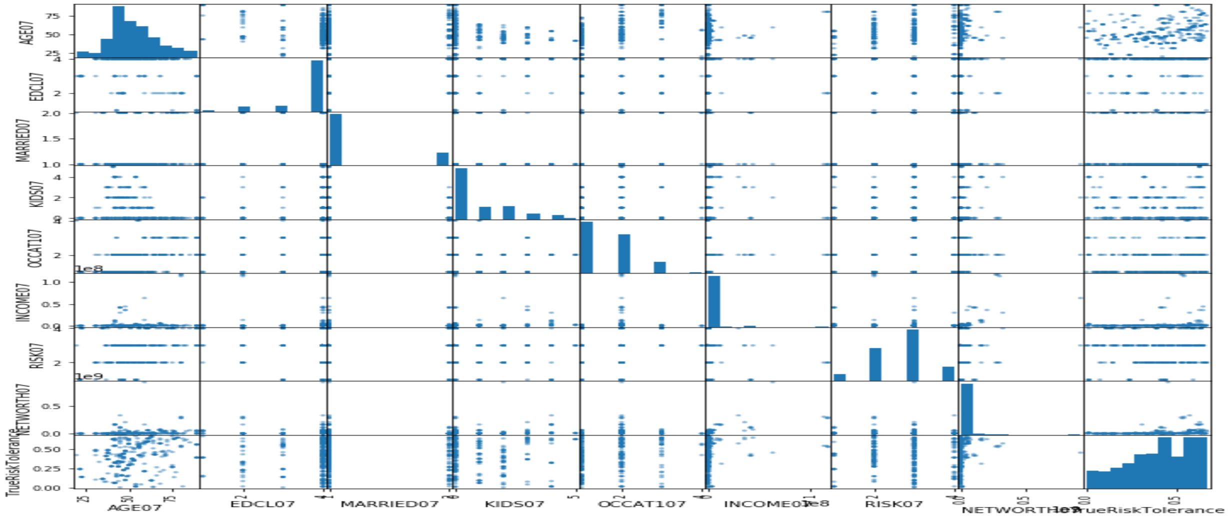
- Spending versus income
- Net worth
- Income

Behavioral attributes

- Willingness to take risk

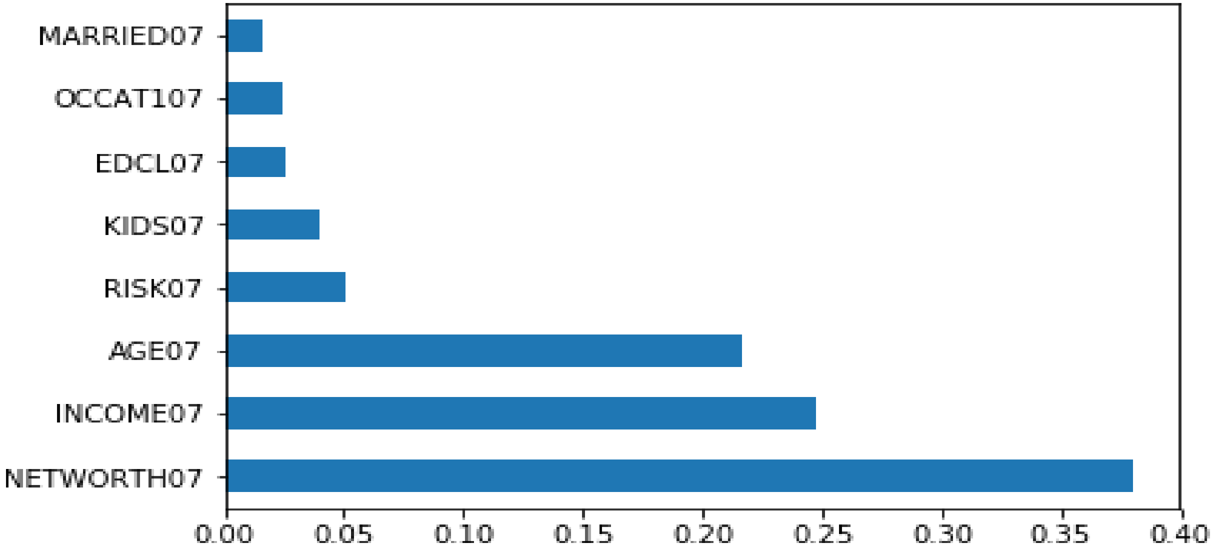
Analysis:

Federal Reserve Board's Survey of Consumer Finances (SCF) data which comprises responses from the same group of people in 2007 (pre-crisis) and 2009 (post-crisis).



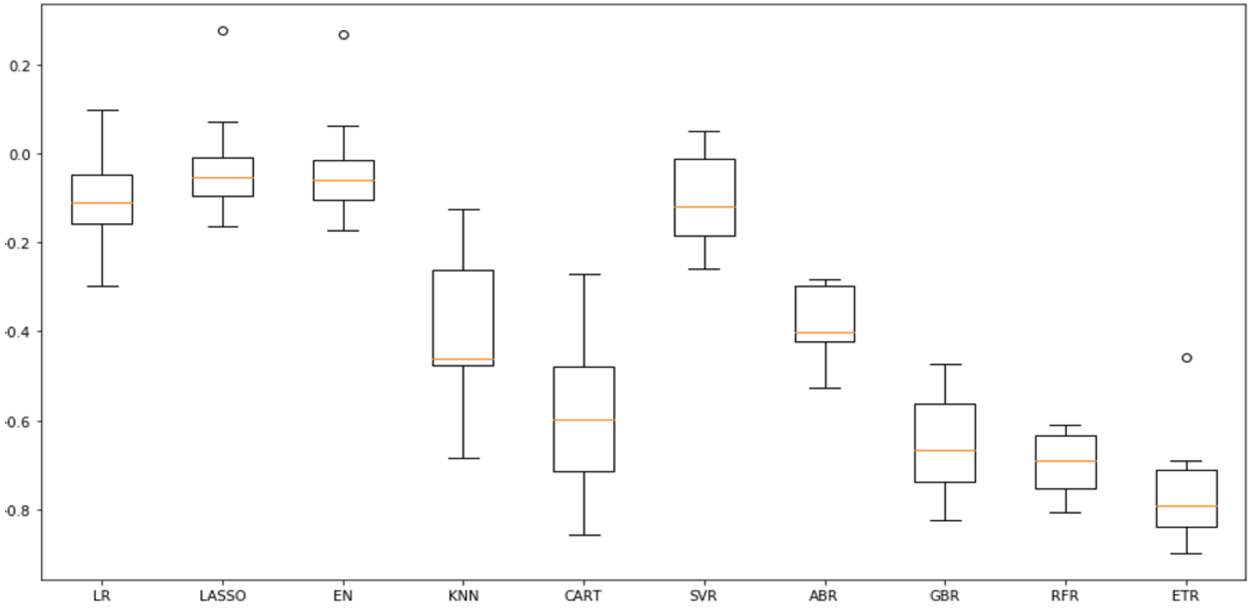
Procedure:

Retaining only the intuitive components from 2007, and exclude all intermediate features and features from 2009, as the variables from 2007 are the only ones needed to forecast risk tolerance.



Results

Ensemble methods such as AdaBoost (boosting) and Random Forest (bagging) has best performance with 0.86 and 0.70 mean squared error respectively.



1.1 Project Overview

According to studies, investors suffer from behavioral biases and are poor assessors of their own risk perception, especially during stressful markets. Furthermore, because these questionnaires must be filled out by hand by investors, they preclude the entire investment management process from being automated.

The archival documents in this case study are taken from the Federal Reserve Board's Survey of Consumer Finances (SCF). The poll comprises responses from the same group of people in 2007 (pre-crisis) and 2009 (post-crisis) on household demographics, net worth, financial, and nonfinancial assets (postcrisis). This enables us to evaluate how each household's allocation changed during global financial crisis of 2008.

The overarching goal is to show how machine learning may be used to automate the manual phases in the portfolio management process.

1.2 Preprocessing

The dataset from "Survey of Consumer Finances"[1] contains the Household's demographics, net worth, financial and non-financial assets for the same demographics in 2007 (pre-crisis) and 2009 (post-crisis).

The predicted variable in the supervised regression framework used for this review is an individual's "actual" risk tolerance, whereas the predictor variables are an individual's demographic, financial, and behavioral traits.

In the methods to follow, we prepare the predicted variable, which is the True risk tolerance. There are a few ways to determine one's "real" risk tolerance. The objective of this research is to propose a mechanism for implementing machine learning to solve the behavioral finance problem.

The following steps are followed to compute the predicted variables:

1. For every individual in the survey, compute the Risky and Riskless assets. The following are the classifications of risky and riskless assets:

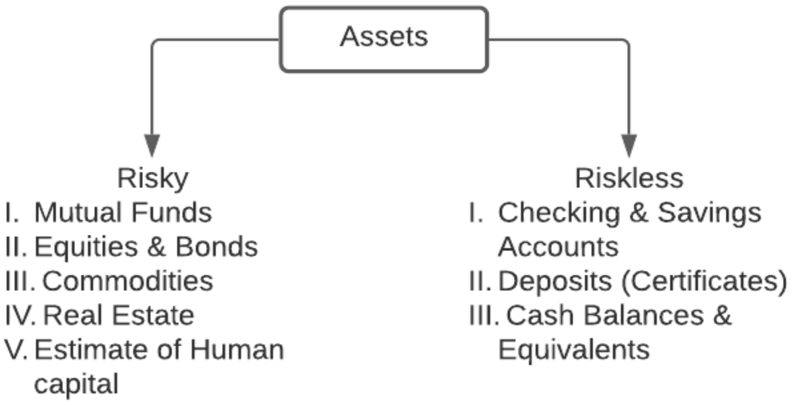

Assets
Risky
I. Mutual Funds
II. Equities & Bonds
III. Commodities
IV. Real Estate
V. Estimate of Human capital
Riskless
I. Checking & Savings Accounts
II. Deposits (Certificates)
III. Cash Balances & Equivalents

Figure 1: Risk based classification of assets

2. We have used the ratio of risky assets to total assets as an indicator of an investor's risk tolerance. We have knowledge on hazardous and riskless assets for people for 2007 and 2009 from the SCF data. To estimate risk tolerance, we analyze this data and normalize the risky assets using the stock price of 2007 vs. 2009.

3. According to a lot of literature, an intelligent investor is one who does not adjust their risk tolerance as the market changes. Consequently, we consider the wise investors to be one of those who changed their risk tolerance by less than 10% between 2007 and 2009. Of course, this is a subjective view that is open to revision.

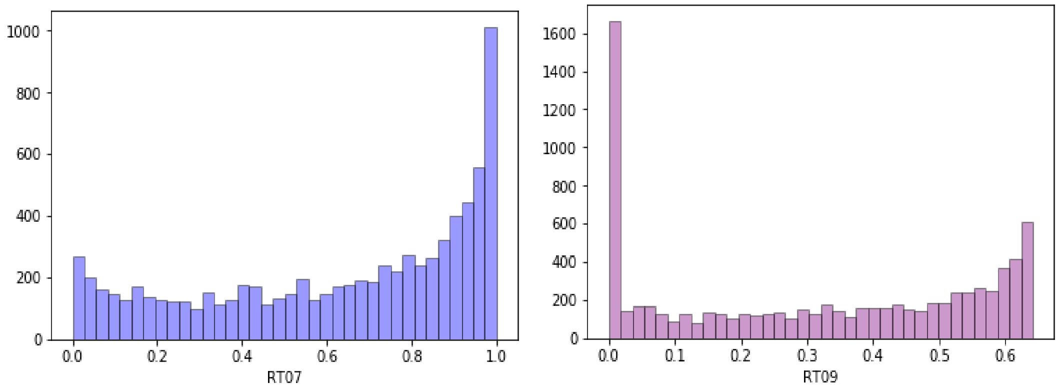


Figure 2. Comparison between risk tolerance for years 2007(pre-crisis) and 2009 (post-crisis).

4. Between 2007 and 2009, we computed the True Risk Tolerance as the average risk tolerance of these 'intelligent' investors. This is the projected variable for our model.
   The goal would be to predict a person's true risk tolerance based on their demographic, financial, and willingness to take risks characteristics.

1.3 Feature Selection

There are more than 500 features[2] in the dataset when seen as a whole. Risk tolerance is highly influenced by investor demographic, financial, and behavioral variables, such as age, current income, net worth, and willingness to take risk, according to academic literature and business practice. These characteristics are used to forecast an investor's risk tolerance.

Retaining only the intuitive components from 2007, and exclude all intermediate features and features from 2009, as the variables from 2007 are the only ones needed to forecast risk tolerance.

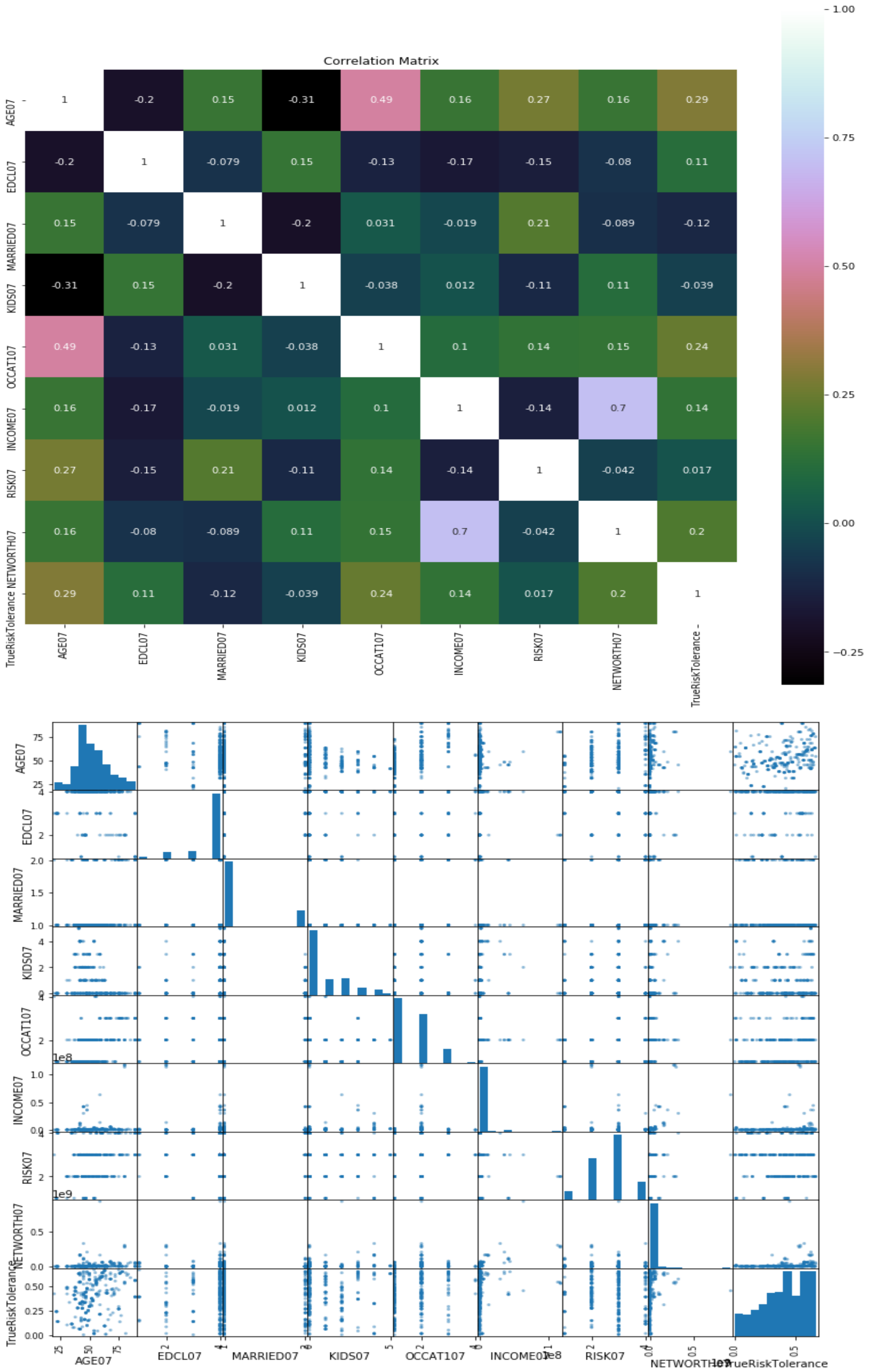


Figure 3: Correlation and scatter matrix for intuitive features. The risk tolerance decreases with factors such as marriage and number of kids. Moreover, there is positive correlation between age and risk tolerance[3].

1.4 Evaluation

We evaluate the performance results based on regression models to capture the relationship between the risk tolerance and the different variables used to predict it. And perform K-fold cross validation and utilize mean squared error score to quantify the results on different models.

Performance details for linear and non-linear models:

1. The nonlinear models perform better than the linear models, which means that there is a nonlinear relationship.
2. Ensemble methods such as AdaBoost (boosting) and Random Forest (bagging) has best performance with 0.86 and 0.70 mean squared error respectively.
3. Furthermore, we fine-tune random forest regressor method using grid-search cv optimization.

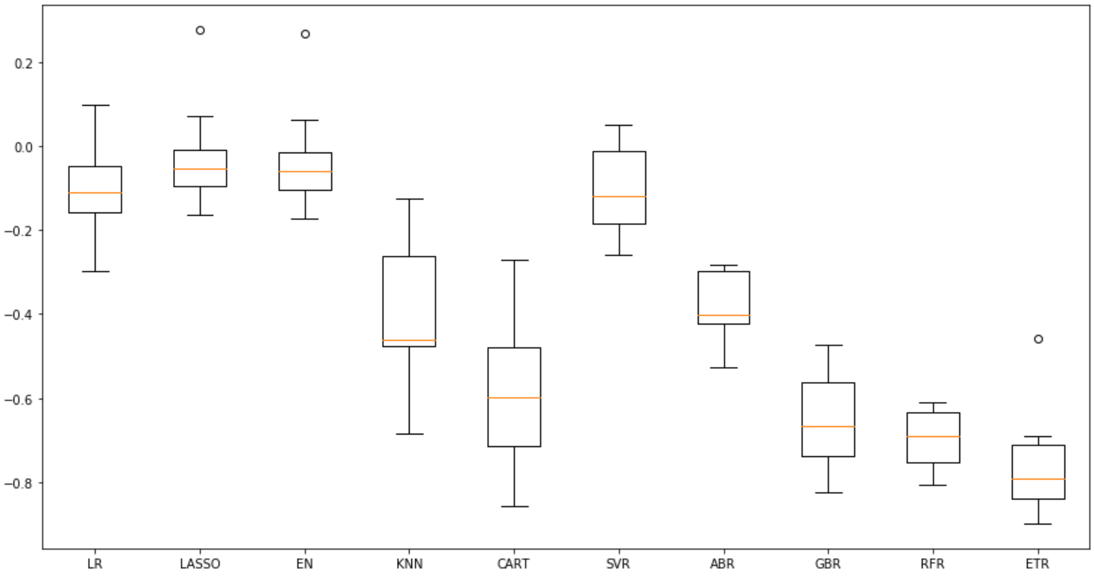


Figure 4: Boxplot performance results on different linear and non-linear models.

Table 1. Results of fine-tuning with respect to number of estimators

Mean Test Score	Standard Deviation	Estimators
0.738	**0.084**	**50**
0.715	0.088	100
0.727	0.085	150
0.735	0.081	200
0.733	0.085	250
0.732	0.082	300
0.728	0.085	350
0.735	0.087	400

4. Using random forest regressor we get 0.757% mean squared error and 0.7678 r2 score for the test set.

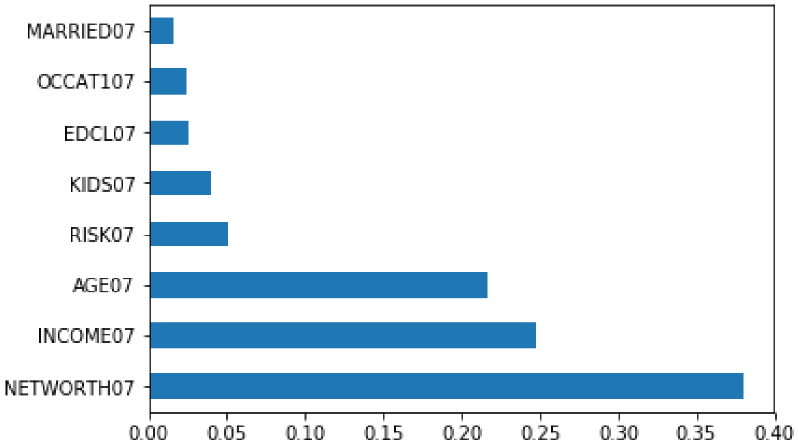


Figure 5: Feature importance based on random forest regression. Income and net worth are the most important factors in determining risk tolerance, followed by age and willingness to accept risks.

1.5 Conclusion

We demonstrated how machine learning models can objectively examine the behavior of different investors in a changing market and attribute these changes to risk appetite-related characteristics. Such models may become more useful if the volume of investor data grows and rich machine learning technology becomes available.

We discovered that the factors and risk tolerance have a non-linear connection. The fundamental variables that determine risk tolerance are income and net worth, followed by age and willingness to accept risks. These characteristics have been used as critical factors to model risk tolerance in several studies.

1.6 References

1. https://www.federalreserve.gov/econres/scf_2009p.htm
2. https://www.federalreserve.gov/econres/files/codebk2009p.txt
3. Wang, Hui & Hanna, Sherman. (1998). Does Risk Tolerance Decrease With Age? Financial Counseling and Planning. 8. 10.2139/ssrn.95489.

Risk Profiling and Tolerance for the Investor

A machine learning based Portfolio management process, by cutting the client completely out of the loop.

- Sidharth Purohit (sp6365)

-Ansh Desai (asd9717)

An algorithm to create a personality profile for the client that better reflects how they would respond to various market circumstances.

Studies have demonstrated that the existing **risk tolerance questionnaires** used in Risk Management processes are prone to inaccuracy, as investors are subject to behavioral biases and are poor judges of their own risk perception, particularly during times of market stress. Furthermore, because these questionnaires must be filled out by hand by investors, they preclude the entire investment management process from being automated.

The overarching goal is to show how machine learning may be used to automate the manual phases in the portfolio management process. Machine learning is also being used to quantify and model investor/individual behavioral bias.

Problem Statement

Problem Description

The predicted variable in the supervised regression framework used for this report is an individual's "real" risk tolerance, while the predictor variables are an individual's demographic, financial, and behavioral traits.

The information utilized in this case study comes from the Federal Reserve Board's Survey of Consumer Finances (SCF). The poll comprises responses from the same group of people in 2007 (pre-crisis) and 2009 (post-crisis) on household demographics, net worth, financial, and non-financial assets (post-crisis). This allows us to examine how each household's allocation changed during the global financial crisis of 2008.

For this case study the data used is from survey of Consumer Finances which is conducted by the Federal Reserve Board. The data source is : https://www.federalreserve.gov/econres/scf_2009p.htm

Getting started

```
import numpy as np
import pandas as pd

##import pandas_datareader.data as web
import matplotlib.pyplot as plt
from pandas.plotting import scatter_matrix
import seaborn as sns
```

```python
import copy
from sklearn.preprocessing import StandardScaler

from sklearn.model_selection import train_test_split
from sklearn.model_selection import KFold
from sklearn.model_selection import cross_val_score
from sklearn.model_selection import GridSearchCV

from sklearn.linear_model import LinearRegression
from sklearn.linear_model import Lasso
from sklearn.linear_model import ElasticNet
from sklearn.tree import DecisionTreeRegressor
from sklearn.neighbors import KNeighborsRegressor
from sklearn.svm import SVR
from sklearn.ensemble import RandomForestRegressor
from sklearn.ensemble import GradientBoostingRegressor
from sklearn.ensemble import ExtraTreesRegressor
from sklearn.ensemble import AdaBoostRegressor
from sklearn.neural_network import MLPRegressor

#Libraries for Deep Learning Models
from keras.models import Sequential
from keras.layers import Dense
from keras.optimizers import SGD
from keras.layers import LSTM
from keras.wrappers.scikit_learn import KerasRegressor

#Libraries for Statistical Models
import statsmodels.api as sm

#Libraries for Saving the Model
from pickle import dump
from pickle import load
```

```
C:\Anaconda2\envs\py36\lib\site-packages\statsmodels\tools\
_testing.py:19: FutureWarning: pandas.util.testing is deprecated. Use
the functions in the public API at pandas.testing instead.
  import pandas.util.testing as tm
Using TensorFlow backend.
```

```python
from urllib.parse import urlencode
```

```python
#ignoring all the warnings
import warnings
warnings.filterwarnings('ignore')
```

Preparing the predicted variable

Household demographics, net worth, financial and non-financial assets for the same demographics in 2007 (pre-crisis) and 2009 are included in the dataset from "Survey of Consumer Finances" (post-crisis).

In the next phases, we prepare the anticipated variable, which is the **True risk tolerance**. There are several methods for determining one's "real" risk tolerance. The goal of this dossier is to develop a method for applying machine learning to solve the behavioral finance problem.

The following are the steps to compute the predicted variables:

1) For each individual in the survey data, compute the **Risky asset** and the **Risk-less asset**. The following are the definitions of risky and risk free assets:

Investments in mutual funds, equities, bonds, commodities, and real estate, as well as an estimate of human capital, are considered "risky assets."

Checking and savings account balances, certificates of deposit, and other cash balances and equivalents are all examples of risk-free assets.

Risky assets is investments in mutual funds, stocks, bonds, commodities, and real estate, and an estimate of human capital.

Risk Free Assets: checking and savings balances,certificates of deposit, and other cash balances and equivalents.

Risk Tolerance is the ratio of risky assets to total assets of an investor and consider that as a measure of risk tolerance of an investor.

2) We use the risky asset to total asset ratio of an investor as a measure of risk tolerance. We have data on hazardous and riskless assets for people for 2007 and 2009 from the SCF data. To calculate risk tolerance, we analyze this data and normalize the hazardous assets using the stock price of 2007 vs. 2009.

Risk Tolerance is simply defined as the ratio of risky assets to riskless assets, normalized using the S&P500 average from 2007 to 2009.

```
Average S&P500 in 2007: 1478
Average S&P500 in 2009: 948
```

3) According to a lot of literature, an intelligent investor is one who does not adjust their risk tolerance as the market changes. As a result, we consider the clever investors to be those who changed their risk tolerance by less than 10% between 2007 and 2009. Of course, this is a subjective assessment that is subject to revision. However, as previously stated, the goal of this case study is to highlight the use of machine learning in behavioral finance and portfolio management and to give a machine learning-based framework that can be further leveraged for more extensive analysis.

```
dataset = pd.read_excel('SCFP2009panel.xlsx')
```

```
dataset

        YY1     Y1         WGT09  AGE07  AGECL07  EDUC07  EDCL07
MARRIED07  \
0         1     11  11668.134198     47        3      12       2
1
1         1     12  11823.456494     47        3      12       2
1
2         1     13  11913.228354     47        3      12       2
1
3         1     14  11929.394266     47        3      12       2
1
4         1     15  11917.722907     47        3      12       2
1
...     ...    ...           ...    ...      ...     ...     ...
...
19280  4423  44231   8170.922241     50        3      16       4
2
19281  4423  44232   8261.846998     50        3      16       4
2
19282  4423  44233   8302.225025     50        3      16       4
2
19283  4423  44234   8335.386179     50        3      16       4
2
19284  4423  44235   8282.547273     50        3      16       4
2

       KIDS07  LIFECL07  ...  NHMORTPCT  WAGEINCPCT
BUSSEFARMINCPCT  \
0           0         2  ... -21.052632  -32.931828              0.0

1           0         2  ... -21.052632  -32.931828              0.0

2           0         2  ... -50.000000  -32.931828              0.0

3           0         2  ... -33.333333  -32.931828              0.0

4           0         2  ... -38.596491  -32.931828              0.0

...       ...       ...  ...        ...         ...              ...

19280       0         1  ... -17.333333   -1.355234           -100.0

19281       0         1  ... -14.864865   -1.355234           -100.0

19282       0         1  ... -14.864865   -1.355234           -100.0

19283       0         1  ... -17.333333   -1.355234           -100.0
```

```
19284       0         1  ... -16.216216    0.471521          -100.0


       INTDIVINCPCT  KGINCPCT  SSRETINCPCT  TRANSFOTHINCPCT
PSAVINGPCT  \
0      15939.278937       0.0          0.0              0.0
93.125197
1      15939.278937       0.0          0.0              0.0
93.125197
2      15939.278937       0.0          0.0              0.0
93.125197
3      15939.278937       0.0          0.0              0.0
93.125197
4      15939.278937       0.0          0.0              0.0
93.125197
...             ...       ...          ...              ...         ..
.
19280    -33.183964       0.0          0.0              0.0
15.875118
19281    -75.946227       0.0          0.0              0.0
15.875118
19282    -10.466512       0.0          0.0              0.0
15.875118
19283    -83.964151       0.0          0.0              0.0
15.875118
19284    -18.484436       0.0          0.0              0.0
15.875118

       LEVERAGEPCT   I
0       270.403054  57
1       249.593620  57
2       209.233358  57
3       209.273158  57
4       232.690767  57
...            ...  ..
19280   -17.136606  57
19281   -14.697926  57
19282   -14.805179  57
19283   -17.020697  57
19284   -16.014887  57

[19285 rows x 515 columns]
```

```
#Average stock index for normalizing the risky assets in 2009
Average_SP500_2007=1478
Average_SP500_2009=948

#Risk Tolerance 2007
dataset['RiskFree07']= dataset['LIQ07'] + dataset['CDS07'] +
dataset['SAVBND07'] + dataset['CASHLI07']
```

```
dataset['Risky07'] = dataset['NMMF07'] + dataset['STOCKS07'] +
dataset['BOND07']
dataset['RT07'] = dataset['Risky07']/(dataset['Risky07']
+dataset['RiskFree07'])

#Risk Tolerance 2009
dataset['RiskFree09']= dataset['LIQ09'] + dataset['CDS09'] +
dataset['SAVBND09'] + dataset['CASHLI09']
dataset['Risky09'] = dataset['NMMF09'] + dataset['STOCKS09'] +
dataset['BOND09']
dataset['RT09'] = dataset['Risky09']/(dataset['Risky09']
+dataset['RiskFree09'])*(Average_SP500_2009/Average_SP500_2007)

dataset2 = copy.deepcopy(dataset)
dataset.head()
```

```
   YY1  Y1         WGT09  AGE07  AGECL07  EDUC07  EDCL07  MARRIED07
KIDS07  \
0    1  11  11668.134198     47        3      12       2          1
0
1    1  12  11823.456494     47        3      12       2          1
0
2    1  13  11913.228354     47        3      12       2          1
0
3    1  14  11929.394266     47        3      12       2          1
0
4    1  15  11917.722907     47        3      12       2          1
0

   LIFECL07  ...  TRANSFOTHINCPCT  PSAVINGPCT  LEVERAGEPCT   I
RiskFree07  \
0         2  ...              0.0   93.125197   270.403054  57
7994.813847
1         2  ...              0.0   93.125197   249.593620  57
7994.813847
2         2  ...              0.0   93.125197   209.233358  57
7984.457871
3         2  ...              0.0   93.125197   209.273158  57
7984.457871
4         2  ...              0.0   93.125197   232.690767  57
7994.813847

   Risky07  RT07  RiskFree09  Risky09      RT09
0      0.0   0.0       16000    17000  0.330422
1      0.0   0.0       19000    18000  0.312036
2      0.0   0.0       13000    12000  0.307876
3      0.0   0.0       25000    13000  0.219429
4      0.0   0.0       17000    12000  0.265410

[5 rows x 521 columns]
```

Let us compute the percentage change in risk tolerance between 2007 and 2009.

```
dataset2['PercentageChange'] =
np.abs(dataset2['RT09']/dataset2['RT07']-1)
```

Checking for the rows with null or nan values and removing them.

```
#Checking for any null values and removing the null values'''
print('Null Values =',dataset2.isnull().values.any())
```

```
Null Values = True
```

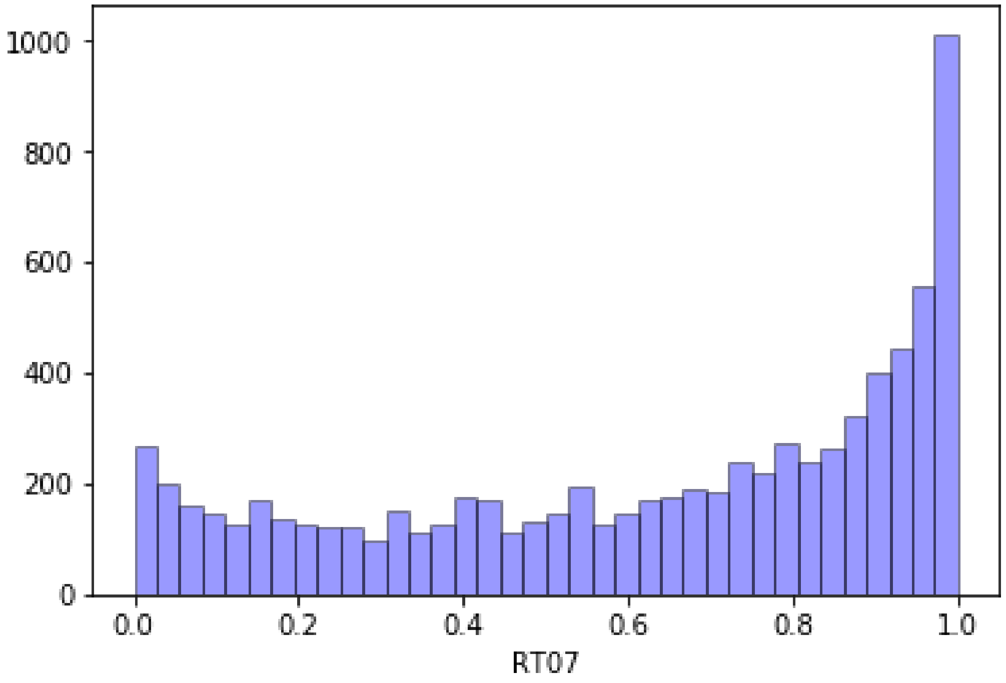
```
# Drop the rows containing NA
dataset2=dataset2.dropna(axis=0)

dataset2=dataset2[~dataset2.isin([np.nan, np.inf, -np.inf]).any(1)]

#Checking for any null values and removing the null values'''
print('Null Values =',dataset2.isnull().values.any())
```

```
Null Values = False
```

Let us plot the risk tolerance of 2007 and 2009.

```
sns.distplot(dataset2['RT07'], hist=True, kde=False,
             bins=int(180/5), color = 'blue',
             hist_kws={'edgecolor':'black'})
```

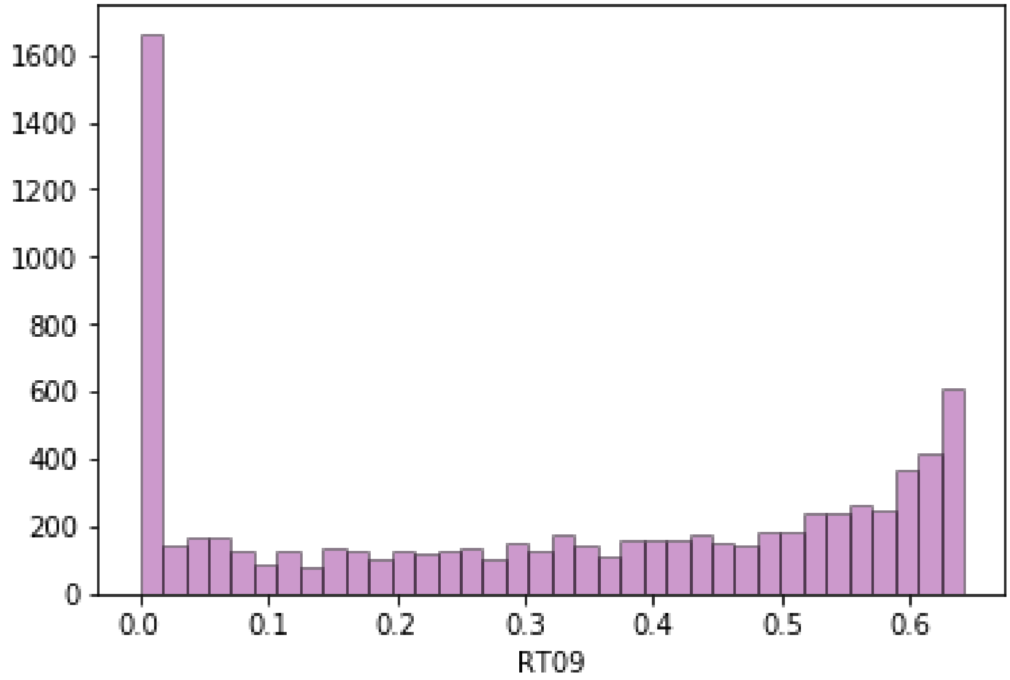
```
<matplotlib.axes._subplots.AxesSubplot at 0x1dd56885860>
```



Looking at the risk tolerance of 2007, we see that a significant number of individuals had risk tolerance close to one. Meaning the investments were more skewed towards the risky assets as compared to the riskless assets. Now let us look at the risk tolerance of 2009.

```
sns.distplot(dataset2['RT09'], hist=True, kde=False,
             bins=int(180/5), color = 'purple',
             hist_kws={'edgecolor':'black'})
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x1dd4899f2e8>
```



Individual's behavior clearly changed after the crisis in 2009, and the majority of their investments were in risk-free assets. Overall risk tolerance dropped in 2009, as evidenced by the fact that the majority of risk tolerance was close to zero.

The next phase is to identify the smart investors whose risk tolerance changed by less than 10% between 2007 and 2009.

We'll now choose the **Intelligent Investors**.

```
dataset3 = copy.deepcopy(dataset2)
```

```
dataset3 = dataset3[dataset3['PercentageChange']<=.1]
```

The **True Risk Tolerance** is defined as the *average risk tolerance of these smart investors between 2007 and 2009. This is the dossier's expected variable.

The goal would be to forecast an individual's genuine risk tolerance based on demographic, financial, and risk-taking characteristics.

```
dataset3['TrueRiskTolerance'] = (dataset3['RT07'] +
dataset3['RT09'])/2

##Let us drop other labels which might not be needed for the
prediction.

dataset3.drop(labels=['RT07', 'RT09'], axis=1, inplace=True)
dataset3.drop(labels=['PercentageChange'], axis=1, inplace=True)
```

Feature Selection-Limit the Feature Space

Features elimination

There are more than 500 features in the dataset when seen as a whole. Risk tolerance is highly influenced by investor demographic, financial, and behavioral variables, such as age, current income, net worth, and willingness to take risk, according to academic literature and business practice. All of these characteristics were included in the dataset and are listed below. These characteristics are used to forecast an investor's risk tolerance.

To refine the characteristics even more, we do the following:

Only preserve the characteristics that are intuitive, as described in the Data Dictionary (https://www.federalreserve.gov/econres/files/codebk2009p.txt, https://www.federalreserve.gov/econresdata/scf/files/fedstables.macro.txt). The following is the description:

1. AGE: There are six age groups, with 1 representing age under 35 and 6 representing age over 75.

2. EDUCATION: There are four education classifications, with 1 denoting no high school and 4 denoting a college diploma.

3. MARRIED: It refers to a couple's marital status. There are two categories, with 1 denoting married and 2 denoting single.

4. OCCU: This abbreviation stands for occupation category. 1 denotes the management category, whereas 4 denotes the unemployed.

5. KIDS: This is the number of children.

6. The NWCAT stands for net worth category. There are five categories, with one representing net value less than 25% and five representing net worth greater than 90%.

7. INCCL: This abbreviation stands for "income category." There are five categories, with one representing a net worth of less than 10,000 dollars and the other representing a net worth of more than 100,000 dollars.

8. RISK: On a scale of 1 to 4, it shows the readiness to take a risk, with 1 representing the highest level of willingness to take a risk.

Keep only the intuitive components from 2007, and exclude all intermediate features and features from 2009, as the variables from 2007 are the only ones needed to forecast risk tolerance.

```
keep_list2 =
['AGE07','EDCL07','MARRIED07','KIDS07','OCCAT107','INCOME07','RISK07',
'NETWORTH07','TrueRiskTolerance'
]
drop_list2 = [col for col in dataset3.columns if col not in
keep_list2]

dataset3.drop(labels=drop_list2, axis=1, inplace=True)

dataset3
```

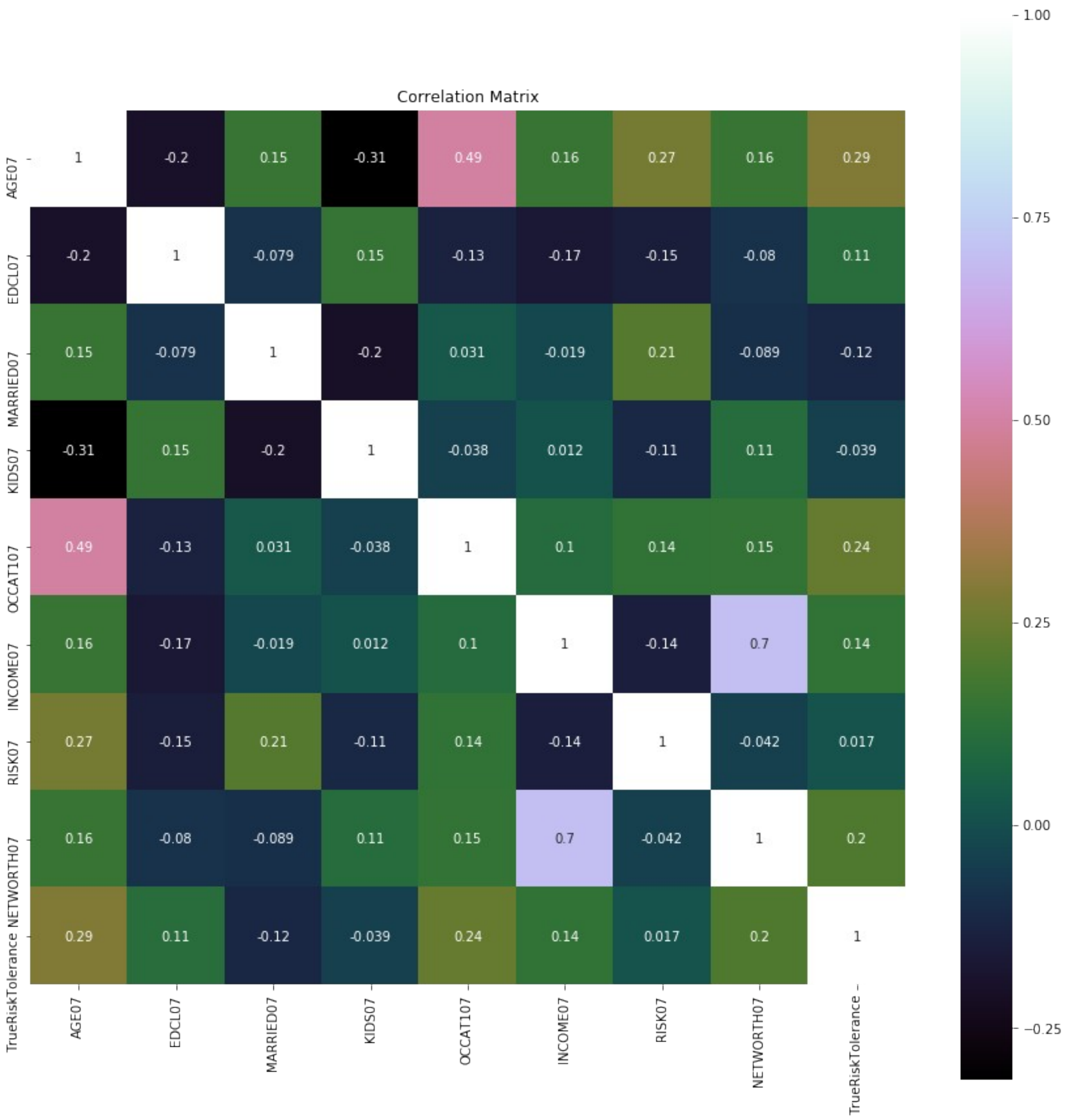
```
       AGE07  EDCL07  MARRIED07  KIDS07  OCCAT107      INCOME07
RISK07  \
60        77       2          1       0         3  3.141680e+04
4
425       55       4          1       1         2  2.779588e+06
2
1122      85       4          1       0         2  3.727417e+05
4
1190      40       2          1       3         1  5.324882e+04
3
1228      70       2          1       0         2  3.716767e+04
3
...      ...     ...        ...     ...       ...           ...     ..
.
19190     53       4          1       0         1  1.810460e+05
2
19191     53       4          1       0         1  1.821109e+05
2
19192     53       4          1       0         1  1.810460e+05
2
19193     53       4          1       0         1  1.810460e+05
2
19194     53       4          1       0         1  1.821109e+05
2

         NETWORTH07  TrueRiskTolerance
60     2.152490e+05           0.199511
425    4.964759e+07           0.641458
1122   5.837768e+07           0.589943
1190   2.688929e+05           0.434127
1228   2.015066e+06           0.228218
...             ...                ...
19190  7.580575e+05           0.352094
19191  7.570219e+05           0.357442
19192  7.580575e+05           0.352094
19193  7.580575e+05           0.352094
```

```
19194  7.580575e+05          0.352094

[355 rows x 9 columns]
```

Let us look at the correlation among the features.

```
# correlation
correlation = dataset3.corr()
plt.figure(figsize=(15,15))
plt.title('Correlation Matrix')
sns.heatmap(correlation, vmax=1,
square=True,annot=True,cmap='cubehelix')
```

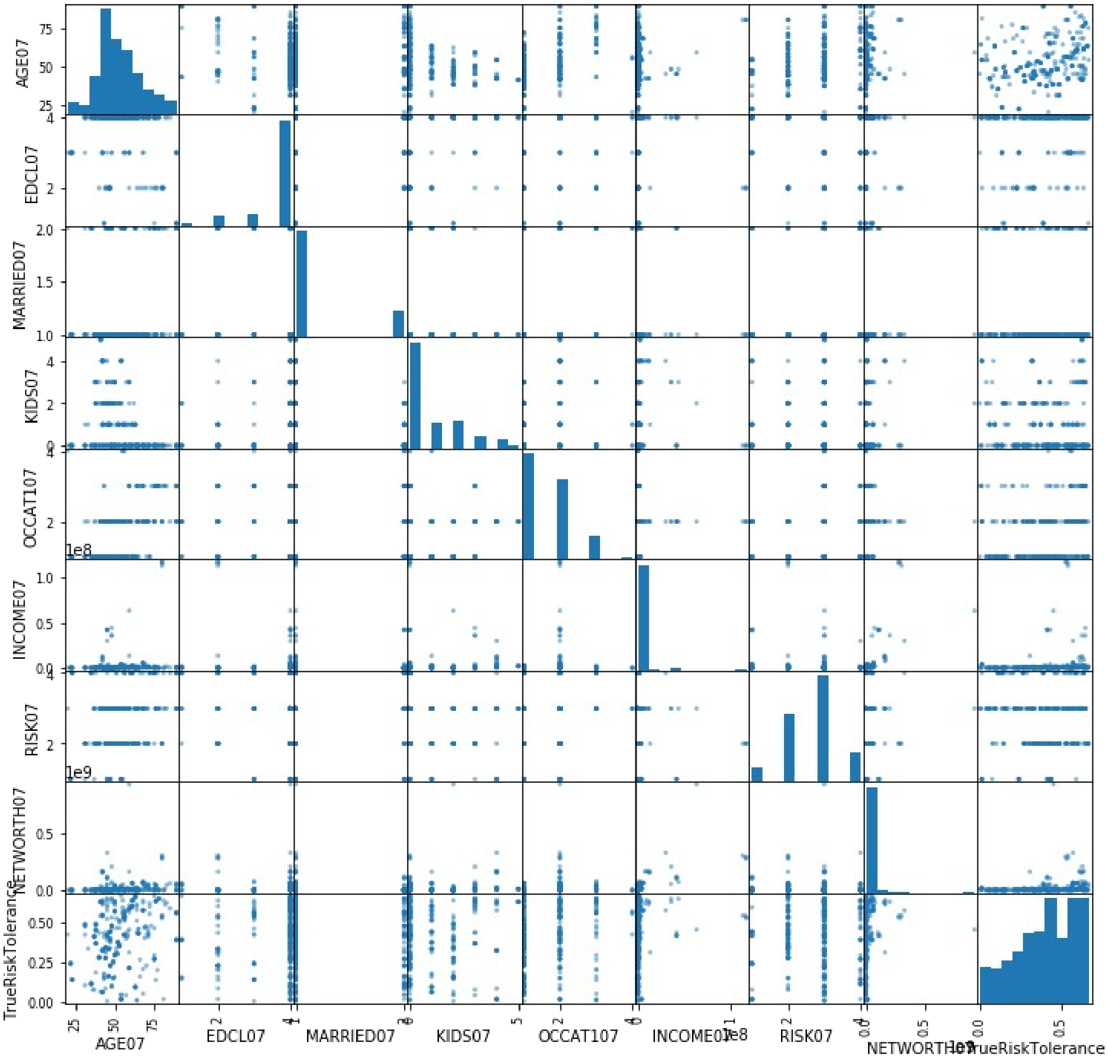
```
<matplotlib.axes._subplots.AxesSubplot at 0x1dd43868748>
```



	AGE07	EDCL07	MARRIED07	KIDS07	OCCAT107	INCOME07	RISK07	NETWORTH07	TrueRiskTolerance
AGE07	1	-0.2	0.15	-0.31	0.49	0.16	0.27	0.16	0.29
EDCL07	-0.2	1	-0.079	0.15	-0.13	-0.17	-0.15	-0.08	0.11
MARRIED07	0.15	-0.079	1	-0.2	0.031	-0.019	0.21	-0.089	-0.12
KIDS07	-0.31	0.15	-0.2	1	-0.038	0.012	-0.11	0.11	-0.039
OCCAT107	0.49	-0.13	0.031	-0.038	1	0.1	0.14	0.15	0.24
INCOME07	0.16	-0.17	-0.019	0.012	0.1	1	-0.14	0.7	0.14
RISK07	0.27	-0.15	0.21	-0.11	0.14	-0.14	1	-0.042	0.017
NETWORTH07	0.16	-0.08	-0.089	0.11	0.15	0.7	-0.042	1	0.2
TrueRiskTolerance	0.29	0.11	-0.12	-0.039	0.24	0.14	0.017	0.2	1

```
# Scatterplot Matrix
from pandas.plotting import scatter_matrix
plt.figure(figsize=(15,15))
scatter_matrix(dataset3,figsize=(12,12))
plt.show()
```

```
<Figure size 1080x1080 with 0 Axes>
```



Net worth and income are positively connected with risk tolerance, as shown in the graph above. The risk tolerance lowers as the number of children and marriage increases. Risk tolerance lowers when one's willingness to take risks declines. There is a positive association between risk tolerance and age.

When other variables are held constant, relative risk aversion diminishes as people age (i.e., the proportion of net wealth invested in hazardous assets grows as people age), according to the paper "Does Risk Tolerance Decrease With Age?" (Hui Wang1,Sherman Hanna). As a result, risk tolerance rises with age.

In conclusion, all of the variables and their link to risk tolerance appear to be intuitive.

Evaluate Algorithms and Models

Let us evaluate the algorithms and the models.

```
# split out validation dataset for the end
Y= dataset3["TrueRiskTolerance"]
X = dataset3.loc[:, dataset3.columns != 'TrueRiskTolerance']
# scaler = StandardScaler().fit(X)
# rescaledX = scaler.transform(X)
validation_size = 0.2
seed = 3
X_train, X_validation, Y_train, Y_validation = train_test_split(X, Y,
test_size=validation_size, random_state=seed)
```

```
# test options for regression
num_folds = 10
#scoring = 'neg_mean_squared_error'
#scoring ='neg_mean_absolute_error'
scoring = 'r2'
```

Compare Models and Algorithms

Regression Models

```
# spot check the algorithms
models = []
models.append(('LR', LinearRegression()))
models.append(('LASSO', Lasso()))
models.append(('EN', ElasticNet()))
models.append(('KNN', KNeighborsRegressor()))
models.append(('CART', DecisionTreeRegressor()))
models.append(('SVR', SVR()))
#Neural Network
#models.append(('MLP', MLPRegressor()))
#Ensable Models
# Boosting methods
models.append(('ABR', AdaBoostRegressor()))
models.append(('GBR', GradientBoostingRegressor()))
# Bagging methods
models.append(('RFR', RandomForestRegressor()))
models.append(('ETR', ExtraTreesRegressor()))
```

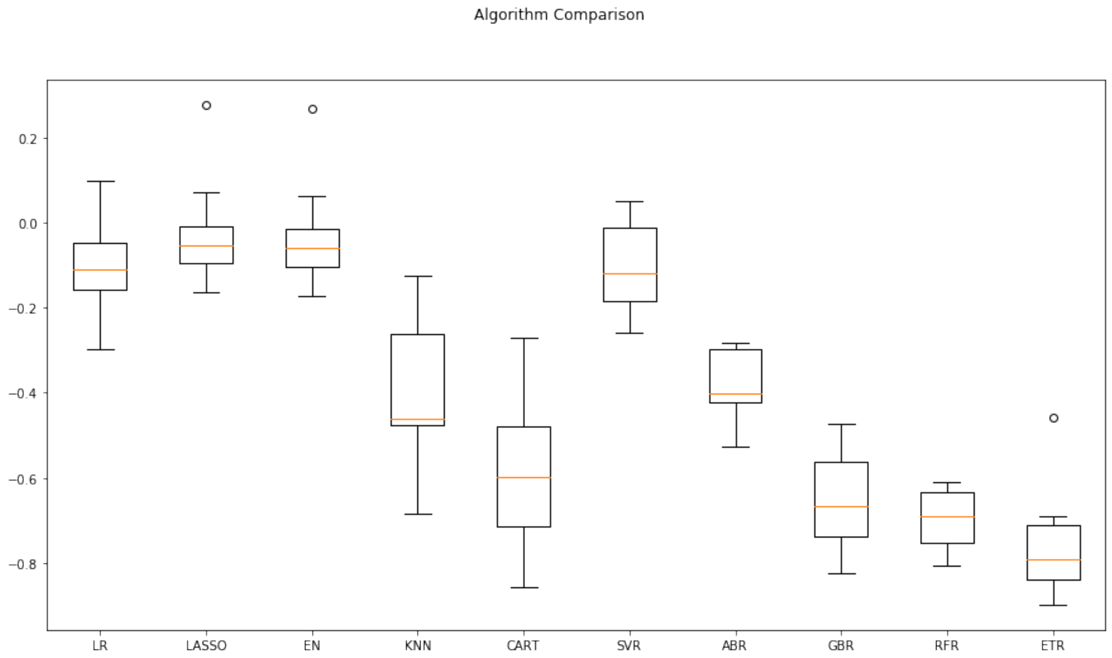
K-folds cross validation

```
results = []
names = []
for name, model in models:
    kfold = KFold(n_splits=num_folds, random_state=seed)
    #converted mean square error to positive. The lower the beter
    cv_results = -1* cross_val_score(model, X_train, Y_train,
cv=kfold, scoring=scoring)
    results.append(cv_results)
    names.append(name)
    msg = "%s: %f (%f)" % (name, cv_results.mean(), cv_results.std())
    print(msg)
```

```
LR: -0.105127 (0.103969)
LASSO: -0.023692 (0.118324)
EN: -0.031108 (0.117770)
KNN: -0.409457 (0.166871)
CART: -0.594662 (0.171557)
SVR: -0.106180 (0.106362)
ABR: -0.386444 (0.086290)
GBR: -0.651666 (0.104657)
RFR: -0.700821 (0.070667)
ETR: -0.759926 (0.118951)
```

Algorithm comparison

```
# compare algorithms
fig = plt.figure()
fig.suptitle('Algorithm Comparison')
ax = fig.add_subplot(111)
plt.boxplot(results)
ax.set_xticklabels(names)
fig.set_size_inches(15,8)
plt.show()
```



Non linear models outperform linear models, implying that there is a non linear relationship between risk tolerance and the factors used to predict it. We utilize random forest regression for grid search because it is one of the finest methods.

Model Tuning and Grid Search

```
###Given that the Random Forest is the best model, Grid Search is
performed on Random Forest.

'''
n_estimators : integer, optional (default=10)
    The number of trees in the forest.
'''
param_grid = {'n_estimators': [50,100,150,200,250,300,350,400]}
model = RandomForestRegressor()
kfold = KFold(n_splits=num_folds, random_state=seed)
grid = GridSearchCV(estimator=model, param_grid=param_grid,
scoring=scoring, cv=kfold)
grid_result = grid.fit(X_train, Y_train)
print("Best: %f using %s" % (grid_result.best_score_,
grid_result.best_params_))
means = grid_result.cv_results_['mean_test_score']
stds = grid_result.cv_results_['std_test_score']
params = grid_result.cv_results_['params']
for mean, stdev, param in zip(means, stds, params):
    print("%f (%f) with: %r" % (mean, stdev, param))
```

```
Best: 0.738440 using {'n_estimators': 50}
0.738440 (0.084927) with: {'n_estimators': 50}
0.715611 (0.088540) with: {'n_estimators': 100}
0.727756 (0.085025) with: {'n_estimators': 150}
0.735676 (0.081626) with: {'n_estimators': 200}
0.733013 (0.085490) with: {'n_estimators': 250}
0.732491 (0.082082) with: {'n_estimators': 300}
0.728702 (0.085524) with: {'n_estimators': 350}
0.735499 (0.087718) with: {'n_estimators': 400}
```

Random forest with number of estimators 50, is the best model after grid search.

Results

```
# prepare model
model = RandomForestRegressor(n_estimators = 250)
model.fit(X_train, Y_train)
```

```
RandomForestRegressor(bootstrap=True, criterion='mse', max_depth=None,
                      max_features='auto', max_leaf_nodes=None,
                      min_impurity_decrease=0.0,
min_impurity_split=None,
                      min_samples_leaf=1, min_samples_split=2,
                      min_weight_fraction_leaf=0.0, n_estimators=250,
                      n_jobs=None, oob_score=False, random_state=None,
                      verbose=0, warm_start=False)
```

```
from sklearn.metrics import r2_score
predictions_train = model.predict(X_train)
print(r2_score(Y_train, predictions_train))
```

```
0.9622238126060929
```

```
# estimate accuracy on validation set
# transform the validation dataset
from sklearn.metrics import mean_squared_error
from sklearn.metrics import r2_score
#rescaledValidationX = scaler.transform(X_validation)
predictions = model.predict(X_validation)

print("Mean square error is :")
print(mean_squared_error(Y_validation, predictions))
print("\n")
print("R2 score is :")
print(r2_score(Y_validation, predictions))
```

```
Mean square error is :
0.007572030007141362



R2 score is :
0.7678811590408414
```

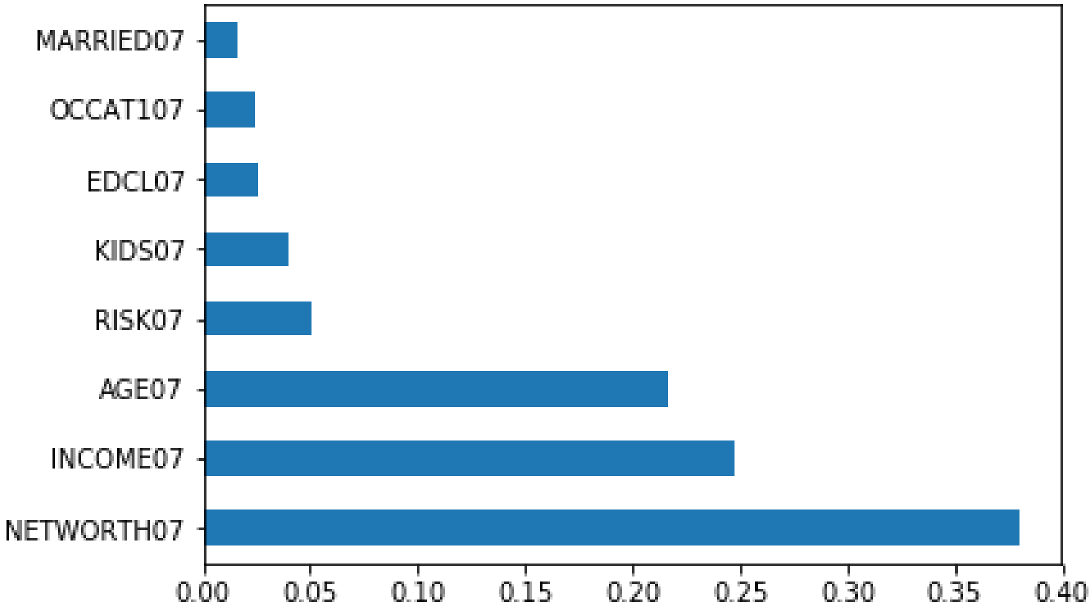
From the mean square error and R2 shown above for the test set, the results look good.

Feature Importance and Features Intuition

Looking at the details above Random forest be worthy of further study. Let us look into the Feature Importance of the RF model

```
import pandas as pd
import numpy as np
model = RandomForestRegressor(n_estimators= 200,n_jobs=-1)
model.fit(X_train,Y_train)
print(model.feature_importances_) #use inbuilt class
feature_importances of tree based classifiers
#plot graph of feature importances for better visualization
feat_importances = pd.Series(model.feature_importances_,
index=X.columns)
feat_importances.nlargest(10).plot(kind='barh')
plt.show()
```

```
[0.21650213 0.02508528 0.01588967 0.03934631 0.02452193 0.247332
 0.0507033  0.38061939]
```



The primary determinants in determining risk tolerance are income and net worth, followed by age and willingness to accept risks, as seen in the graph above. These characteristics have been identified as key variables in the modeling of risk tolerance in various studies.

Save Model for Later Use

```
# Save Model Using Pickle
from pickle import dump
from pickle import load

# save the model to disk
filename = 'finalized_model_dossier-3.sav'
dump(model, open(filename, 'wb'))

# load the model from disk
loaded_model = load(open(filename, 'rb'))
# estimate accuracy on validation set
predictions = loaded_model.predict(X_validation)
result = mean_squared_error(Y_validation, predictions)
print(r2_score(Y_validation, predictions))
print(result)
```

```
0.7594208453021829
0.007848016864710691
```

Conclusion

We demonstrated how machine learning models could objectively examine the behavior of different investors in a changing market and attribute these changes to risk appetite-related characteristics. Such models may become more useful if the number of investment data grows and rich machine learning technology becomes available.

We discovered that the factors and risk tolerance have a non-linear connection. The primary variables that determine risk tolerance are income and net worth, followed by age and willingness to accept risks. These characteristics have been identified as key variables in the modeling of risk tolerance in various studies.