DATA5322 Written Homework 2
Ting-Yu, Lin

Analysis of factors affecting homeownership: A Support Vector Machine Approach

Abstract

This project uses a support vector machine (SVM) model to analyze Washington state housing data, applying both linear and nonlinear SVMs with radial basis functions and polynomial kernels to identify key predictors of homeownership. By adjusting the C parameter, gamma value and degree, the optimal model parameters were found. The test accuracy of all models reached 70%. The analysis found that married people, those aged 55 to 75 and those with a college education were more likely to own property. Additionally, homes with more rooms and bedrooms are more likely to be owner-occupied. The findings reveal a social problem: It is difficult for young people to buy their own homes these days. We hope the government will provide home purchase subsidies or other support measures to help young people overcome the challenges of buying homes.

Introduction

When exploring the question of whether to buy or simply rent a home, we'll find that there are many factors that influence whether a home is for owner-occupation or rental. These factors related to the home as well as related to the individual.

This analysis will explore selected data from Washington State collected through the U.S. Census and accessed through IPUMS USA1. This dataset contains variables about people and their housing situation. In a data set, there are variables such as people's age, income, or education. These housing-related variables include number of rooms, electricity costs, and year of construction, among others. We will conduct classification analysis through SVM to explore the hidden patterns and trends in these data. We will use different SVM kernel functions, including linear, radial basis and polynomial kernel functions, to compare their performance.

Try to find relevant analysis and recommendations on current housing issues. For many people, buying a home is a big decision. Through this analysis, we hope to not only provide in-depth insights into home buying issues, but also provide some valuable advice.

Background

In our project, we use support vector machine model for data analysis. A support vector machine (SVM) is a supervised machine learning algorithm that classifies data by finding a hyperplane that maximizes the distance (margin) between each class in an N-dimensional space.[1] Decision boundary will be affected by support vectors.

Linear SVM

linear SVM separate data with a line, this means that data do not need to undergo any transformations to separate the data into different classes. Mathematically, the separating hyperplane can be represented as:

$$wx + b = 0$$

where w is the weight vector, x is the input vector, and b is the bias term.

The adjustable margin parameter is C; a larger C value will narrow the range of the minimum misclassification limit, while a smaller C value will expand the range, allowing more misclassified data.[1]

Nonlinear SVM

Nonlinear SVMs handle data that isn't linearly separable. To separate data, preprocessing transforms data into higher-dimensional space. Higher dimensional spaces can create more complexity by increasing the risk of overfitting the data and by becoming computationally taxing. The “kernel trick” helps to reduce some of that complexity, making the computation more efficient, and it does this by replacing dot product calculations with an equivalent kernel function.[2] In this project, we will use Radial basis function kernel (RBF kernel) and Polynomial kernel. The formulas as below:

$$K(x_i, x_{i'}) = \exp\left(-\gamma \sum_{j=1}^{p} (x_{ij} - x_{i'j})^2\right)$$

RBF kernel

$$K(x_i, x_{i'}) = \left(1 + \sum_{j=1}^{p} x_{ij} x_{i'j}\right)^d$$

Polynomial kernel

We can adjust the degree parameter in the Polynomial kernel function and the gamma parameter in the RBF kernel function to enhance model performance.

Methodology

Data Preparation

To prepare the data for our models, we initially selected five variables of interest : 'ROOMS'(Number of rooms in the house), 'BEDROOMS'(Number of bedrooms in the house), 'MARST'(Marital status), 'EDUC'(Education status) and 'AGE'. The target variable is 'OWNERSHP' (The status of the house is rental or owner-occupied).

Next, in some variable columns, certain values like 9999999, 999, 99 and 9 represent missing data or lack numerical significance. Therefore, we convert them to NA and remove them.
This dataset contains information about multiple people living in each residence. To facilitate analysis, we retained the person with the highest income in each household (the highest value of 'INCTOT') since they were most likely to be the homeowners, while deleting the information of others.

Hyperplanes must be computed using continuous numerical data. Therefore, we will reclassify 'MARST' into married and single categories and perform dummy encoding to create new variables 'Married' and 'Single'. Similarly, we will reclassify 'EDUC' into those with and without college education and perform dummy encoding to create new variables 'college' and 'no_college'.

The original data set was too large to run, so we randomly selected 5,000 data for analysis.

Models

The data is divided into 70% for training, and the remaining 30% is reserved for testing the accuracy of the model. In the linear SVM model method, we selected 'ROOMS' and 'BEDROOMS' as variables to predict the rental status or owner-occupied status of the house. And adjust parameter C, ranging from 0.001, 0.01, 0.1, 1, 5 to 10. Use cross-validation to find the best parameters. The best-parameter model was then used to fit the test data to determine its accuracy.

In the nonlinear SVM model method with radial basis function kernel, we selected 'AGE', 'Married' and 'Single' as variables to predict the rental status or owner-occupied status of the house. We adjust parameter C, ranging from 0.01, 0.1, 1, 5 to 10 and adjust garmma from 0.5,1,2,3 to 4. Use cross-validation to identify the optimal

parameters. The best-parameter model was then used to fit the test data to determine its accuracy.

For the nonlinear SVM model method with polynomial kernel, we selected 'AGE', 'Married' and 'Single' as variables to predict the rental status or owner-occupied status of the house. We adjust parameter C, ranging from 0.01, 0.1, 1, 5 to 10 and adjust degree from 1,2 to 3. Similarly, cross-validation was used to identify the optimal parameters, and the best-parameter model was applied to the test data to determine its accuracy.

Results

Figure 1 shows that houses with more rooms and bedrooms tend to be owned. We utilized linear SVM to train on training data, selecting 'ROOMS' and 'BEDROOMS' as variables to predict the rental status or owner-occupied status of the house. We adjusted the parameter C and used cross-validation to find the optimal parameters. The results indicate that C equals 0.01 is the best parameter value, with a mean test score of 0.8116. We applied the best-parameter model to fit the test data, as shown in Figure 2. The accuracy is 0.828, with 1776 support vectors. Visualizing by plotting values of the support vector classifier decision function, Figure 3 shows that the model can mostly correctly distinguish between the two classes, with only a few misclassifications. Furthermore, with a smaller value of the C parameter, we obtain a larger number of support vectors because the margin is now wider.

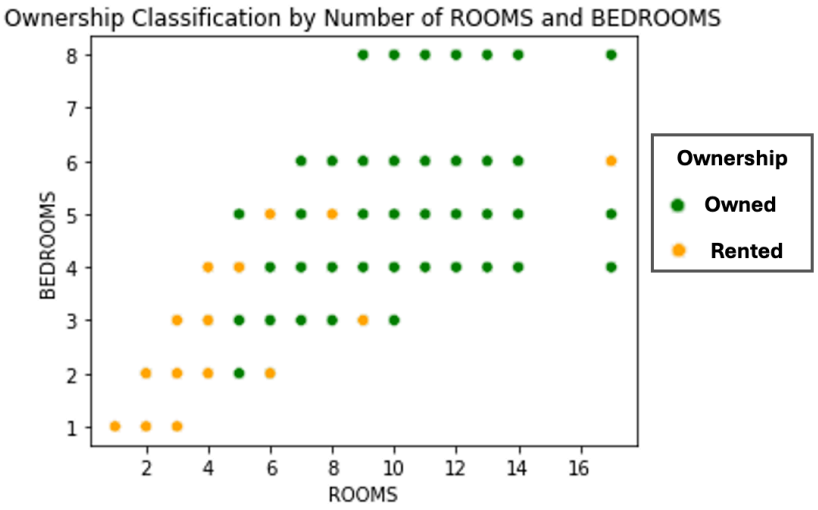

Figure 1 The scatter plot showing the distribution of 'ROOMS' and 'BEDROOMS', with different colors indicating ownership status.

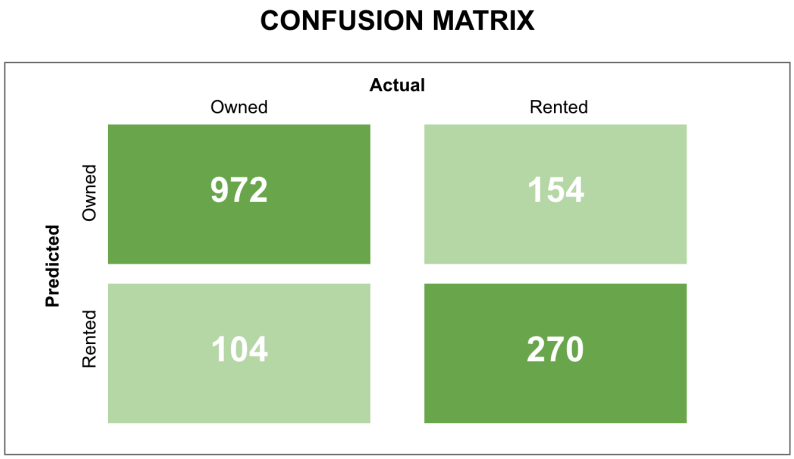

Figure 2 The confusion matrices for linear SVM.

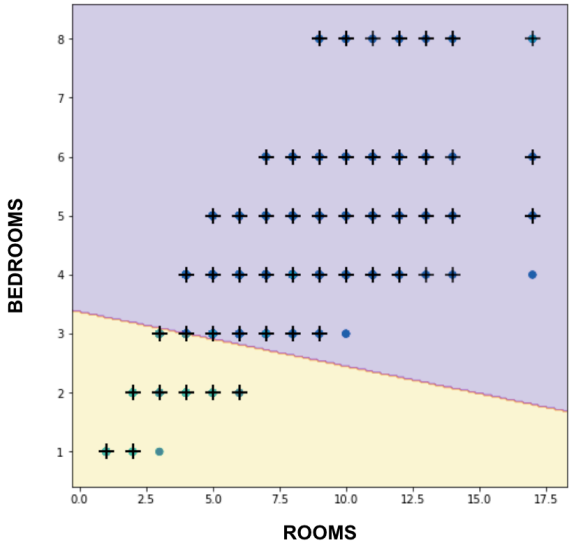

Figure 3 Scatter Plot of Bedroom numbers vs. Room numbers by Ownership Status with decision line.

Figure 4 depicts the age distribution according to ownership and marital status. We can observe that the age distribution of homeowners has relative peaks ranging from 55 to 75 years old. There are more single renters than married renters. We selected 'AGE', 'Married' and 'Single' as variables to predict the rental status or owner-occupied status of the house using RBF kernel SVM model. Turning C parameter and gamma value, and vaildate with cross vaildation. The result shows as Table 1. The best parameters are C equals 1 and gamma equals 0.5, can get 0.736286 mean test score. We applied the best-parameter model to fit the test data, as shown in Figure 5. The accuracy is 0.750, with 1919 support vectors.

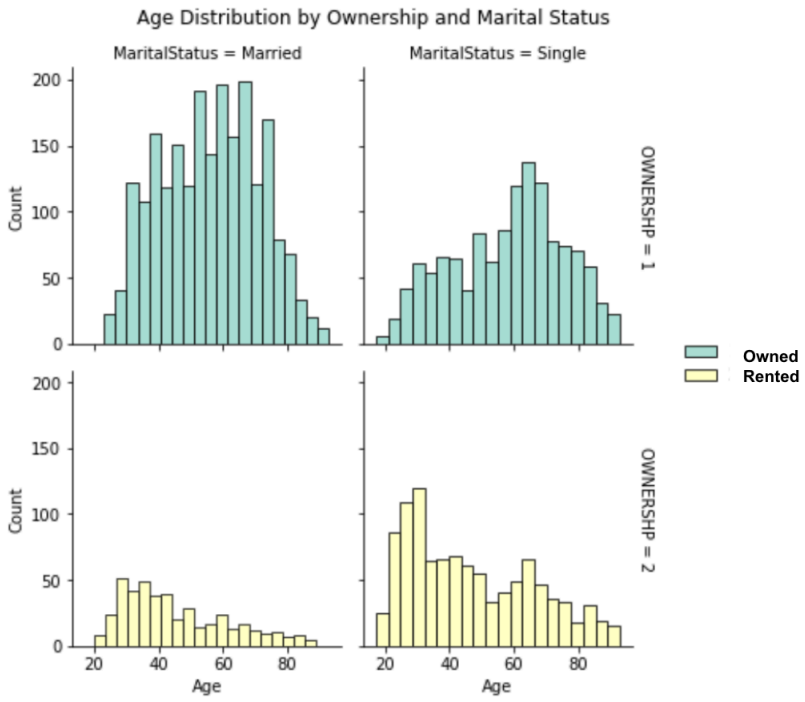

Figure 4 Histogram of age distribution by ownership and marital status.

C value / gamma	0.01	0.1	1	5	10
0.5	0.69914286	0.727143	0.736286	0.730857	0.729714
1	0.69914286	0.727143	0.731429	0.730857	0.728857
2	0.69914286	0.717429	0.731429	0.729429	0.728857
3	0.69914286	0.716571	0.731429	0.729429	0.729429
4	0.69914286	0.713143	0.731429	0.729429	0.729429

Table 1 Results of the best parameters for the SVM model with RBF kernel.

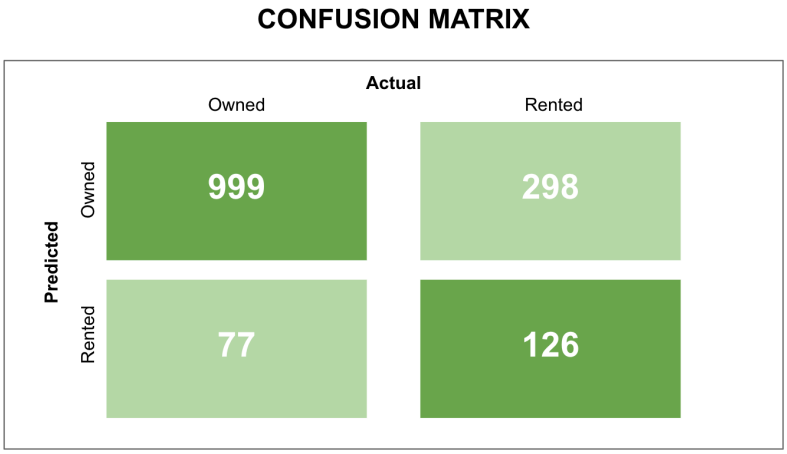

Figure 5 The confusion matrices for nonlinear SVM with RBF kernel.

Figure 6 shows the age distribution according to ownership and education status. We can observe the similar result as Figure 4 that the age distribution of homeowners has relative peaks ranging from 55 to 75 years old. Additionally, college-educated individuals are more likely to own a house compared to those who haven't attended college. We selected 'AGE', 'college', and 'no_college' as variables to predict the rental status or owner-occupied status of the house using a polynomial kernel SVM model. We adjusted the C parameter and degree and validated them using cross-validation. The results are presented in Table 2, where the mean test score is consistently 0.699143. Therefore, we chose C equals 0.01 and degree equals 1 as the optimal parameters. We applied the best-parameter model to fit the test data, as in Figure 7. The accuracy is 0.717, with 2106 support vectors.

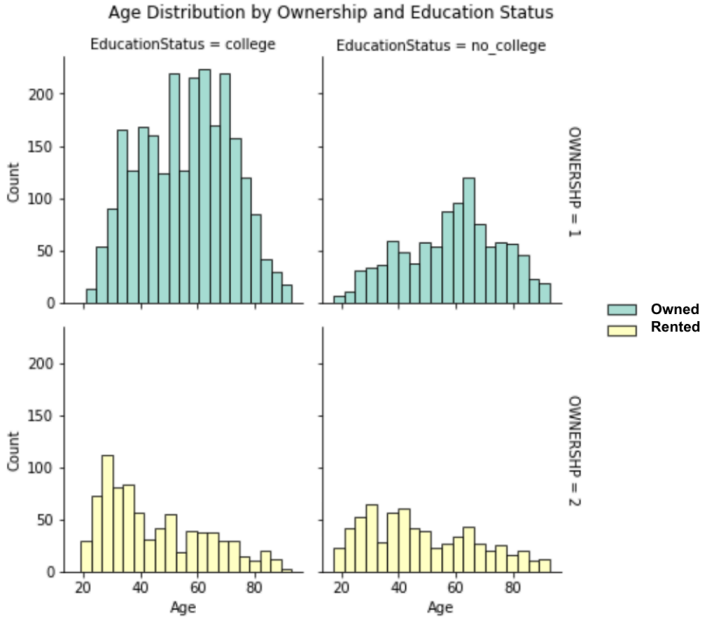

Figure 6 Histogram of age distribution by ownership and education status.

C value / degree	0.01	0.1	1	5	10
1	0.699143	0.699143	0.699143	0.699143	0.699143
2	0.699143	0.699143	0.699143	0.699143	0.699143
3	0.699143	0.699143	0.699143	0.699143	0.699143

Table 2 Results of the best parameters for the SVM model with polynomial kernel.

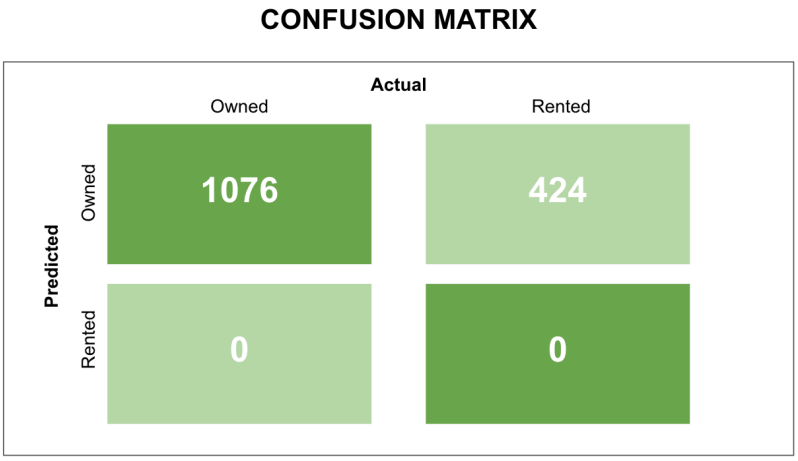

Figure 7 The confusion matrices for nonlinear SVM with polynomial kernel.

Discussion

From the analysis part of the linear SVM model, it can be seen that the number of bedrooms and rooms appears to be a strong predictor of ownership status with approximately 83% accuracy. The more bedrooms and rooms a property has, the more likely it is that it will be owner-occupied. The number of bedrooms is 3 to 4 as the dividing point. Properties with a total of 1 or 2 rooms and bedrooms are mostly used for rental purposes. Such properties often lack space beyond bedrooms, which can be an important consideration for potential homebuyers.

The analysis of the household's age, marital status and education level shows that using the nonlinear support vector machine (SVM) model for analysis can achieve an accuracy of more than 70%. This indicates that age, marital status and education are also reliable factors to analyze. Most people who can own their own homes are between the ages of 55 and 75. In this group of age may be able to accumulate enough funds to

purchase a home. This also shows a social problem, that is, housing prices are too high. It makes impossible for young people to afford mortgages, and they can only live in rented houses. In terms of marital status, more married people buy houses than single people. This may be because after marriage, couples can jointly bear the mortgage loan, making it easier for singles to purchase a house. An analysis of educational attainment shows that people with a college education are more likely to buy homes. People with higher education are more likely than others to find higher-paying jobs and are therefore more able to afford mortgages.

Whether or not to buy a home is an important issue in everyone's life. However, as housing prices continue to rise, the inability of most young people to afford their own home.The government should take action and develop schemes like youth home purchase subsidies to assist young people in buying homes.

Conclusions

This project uses a linear SVM model and a nonlinear SVM model to successfully predict whether a house is owned or rented. The test accuracy of models all can reach more than 70%. The analysis shows that married people, aged 55 to 75 and those with a college education are more likely to own property. Additionally, properties with more rooms and bedrooms are more likely to be owner-occupied.

The research also reveals a deep social problem: in the current economic and social environment, it is increasingly difficult for young people to buy their own homes. We hopes that the government can face this problem and provide young people with home purchase subsidies or other support measures to solve difficulty in purchasing homes for young people.

References

[1]What are support vector machines (SVMs)?, IBM, 2023
[2]Support vector machines, Sontag, David, New York University
[3] Steven Ruggles, Sarah Flood, Matthew Sobek, Danika Brockman, Grace Cooper, Stephanie Richards, and Megan Schouweiler. IPUMS USA: Version 13.0 [dataset]. Minneapolis, MN: IPUMS, 2023.

Appendix

Data Preparation

```
# remove NA data
df_clean = df[df['OWNERSHP'] != 0]
df_clean = df_clean[df_clean['ROOMS'] != 0]
df_clean = df_clean[df_clean['BEDROOMS'] != 0]
df_clean = df_clean[df_clean['INCTOT'] != 9999999]
df_clean = df_clean[df_clean['MARST'] != 9]
df_clean = df_clean[df_clean['AGE'] != 999]
df_clean = df_clean[df_clean['EDUC'] != 99]

# select the highest value of 'INCTOT' of each household and delete the others.
idx = df_clean.groupby('SERIAL')['INCTOT'].idxmax()
df_clean = df_clean.loc[idx]
duplicates = df_clean['SERIAL'].duplicated()
duplicates.any()

# reclassify 'MAST' and create dummy variables
marital_map = {
    1: 'Married',
    2: 'Married'
    3: 'Single',
    4: 'Single',
    5: 'Single',
    6: 'Single',
}
df_clean['MaritalStatus'] = df_clean['MARST'].map(marital_map)
df_clean_dummies = pd.get_dummies(df_clean['MaritalStatus'])
df_clean = pd.concat([df_clean, df_clean_dummies], axis=1)

# reclassify 'EDUC' and create dummy variables
education_map = {
    0: 'no_college',
    1: 'no_college',
    2: 'no_college',
    3: 'no_college',
    4: 'no_college',
    5: 'no_college',
    6: 'no_college',
```

```
    7: 'college',
    8: 'college',
    9: 'college',
    10: 'college',
    11: 'college',
}
df_clean['EducationStatus'] = df_clean['EDUC'].map(education_map)
df_clean_dummies = pd.get_dummies(df_clean['EducationStatus'])
df_clean = pd.concat([df_clean, df_clean_dummies], axis=1)

# reduce sampe size to 5000
df_clean = df_clean.sample(n=5000, random_state=42)

# Split data to train dataset and test dataset
X_train, X_test, y_train, y_test =  train_test_split(X
                                    ,y
                                    ,train_size = 0.7
                                    ,random_state = 1)
```

Linear SVM

```
# variables
X = df_clean[['ROOMS', 'BEDROOMS']] # feature
y= df_clean['OWNERSHP'] # target variable

# model and cross vaildation
svm_linear = SVC(kernel='linear')
svm_linear.fit(X_train, y_train)

kfold = skm.KFold(5,
                  random_state=0,
                  shuffle=True)
grid = skm.GridSearchCV(svm_linear,
                        {'C':[0.001,0.01,0.1,1,5,10]},
                        refit=True,
                        cv=kfold,
                        scoring='accuracy')
grid.fit(X, y)
grid.best_params_
grid.cv_results_[('mean_test_score')]
```

```
# test acuracy
best_ = grid.best_estimator_
y_test_hat = best_.predict(X_test)
print(confusion_table(y_test_hat, y_test))
accuracy = accuracy_score(y_test, y_test_hat)
print("Accuracy:", accuracy)

# support_vectors
svm_linear = SVC(C=0.01,kernel='linear')
svm_linear.fit(X_train, y_train)
num_support_vectors = len(svm_linear.support_)

# figure 1
df['OwnershipStatus'] = df['OWNERSHP'].map({1: 'owned', 2: 'rented'})
palette_colors = {1: "green", 2: "orange"}
sns.scatterplot(x='ROOMS', y='BEDROOMS', hue='OWNERSHP', data=df,
palette=palette_colors)
plt.title('Ownership Classification by Number of ROOMS and BEDROOMS')
plt.xlabel('ROOMS')
plt.ylabel('BEDROOMS')
plt.subplots_adjust(right=0.8)
plt.legend(title='Ownership', loc='center left', bbox_to_anchor=(1, 0.5))
plt.show()

# figure 3
fig, ax = subplots(figsize=(8,8))
plot_svm(X,
        y,
        svm_linear,
        ax=ax)
```

RBF

```
# variables
X = df_clean[['AGE','Married','Single']] # feature
y= df_clean['OWNERSHP'] # target variable

# model and cross vaildation
svm_rbf = SVC(kernel='rbf')
svm_rbf.fit(X_train, y_train)
```

```
kfold = skm.KFold(5,
                  random_state=0,
                  shuffle=True)
grid = skm.GridSearchCV(svm_rbf,
                        {'C':[0.01,0.1,1,5,10], 'gamma':[0.5,1,2,3,4]},
                        refit=True,
                        cv=kfold,
                        scoring='accuracy')
grid.fit(X_train, y_train)
grid.best_params_
grid.cv_results_[('mean_test_score')]

# test acuracy
best_ = grid.best_estimator_
y_test_hat = best_.predict(X_test)
print(confusion_table(y_test_hat, y_test))
accuracy = accuracy_score(y_test, y_test_hat)
print("Accuracy:", accuracy)

# support_vectors
svm_rbf = SVC(C=1, gamma=0.5,kernel='rbf')
svm_rbf.fit(X_train, y_train)
num_support_vectors = len(svm_rbf.support_)
num_support_vectors

# figure 4
g = sns.FacetGrid(df_clean, row='OWNERSHP', col='MaritalStatus',
hue='OWNERSHP', palette='Set3', margin_titles=True)
g.map(sns.histplot, 'AGE', bins=20, kde=False)
g.add_legend()
g.set_axis_labels('Age', 'Count')
g.fig.suptitle('Age Distribution by Ownership and Marital Status', y=1.03)
plt.show()
```

Polynomial

```
# variables
X = df_clean[['AGE','college','no_college']] # feature
y= df_clean['OWNERSHP'] # target variable
```

```
# model and cross vaildation
svm_poly = SVC(kernel='poly')
svm_poly.fit(X_train, y_train)

kfold = skm.KFold(5,
                  random_state=0,
                  shuffle=True)
grid = skm.GridSearchCV(svm_poly,
                        {'C':[0.01,0.1,1,5,10], 'degree':[1,2,3]},
                        refit=True,
                        cv=kfold,
                        scoring='accuracy')
grid.fit(X_train, y_train)
grid.best_params_
grid.cv_results_[('mean_test_score')]

# test acuracy
best_ = grid.best_estimator_
y_test_hat = best_.predict(X_test)
print(confusion_table(y_test_hat, y_test))
accuracy = accuracy_score(y_test, y_test_hat)
print("Accuracy:", accuracy)

# support_vectors
svm_poly = SVC(C=0.01, degree=1,kernel='poly')
svm_poly.fit(X_train, y_train)
num_support_vectors = len(svm_poly.support_)

# figure 6
g = sns.FacetGrid(df_clean, row='OWNERSHP', col='EducationStatus',
hue='OWNERSHP', palette='Set3', margin_titles=True)
g.map(sns.histplot, 'AGE', bins=20, kde=False)
g.add_legend()
g.set_axis_labels('Age', 'Count')
g.fig.suptitle('Age Distribution by Ownership and Education Status', y=1.03)
plt.show()
```