



# PRIRODNO-MATEMATIČKI FAKULTET INFORMATIKA

PREDMET:

Uvod u nauku o podacima

SEMINARSKI RAD NA TEMU:

House Prices - Advanced Regression Techniques

Članovi tima

Tomislav Manojlović 70/2022

Jovan Žarković 54/2022

Predmetni profesor

Branko Arsić

Uvod.....	3
Opis problema i motivacija .....	3
Opis i priprema podataka .....	4
Izvor podataka .....	4
Čišćenje i obrada podataka.....	11
Eksplorativna analiza podataka (EDA) .....	58
Izbor promenljivih / Feature Engineering.....	75
Implementacija i procena modela.....	87
Poređenje modela .....	107
Zaključak .....	108
Literatura .....	109

# Uvod

Predviđanje cena kuća ima veliki značaj u savremenom svetu jer omogućava donošenje boljih ekonomskih i investicionih odluka. Tačne procene vrednosti nekretnina pomažu kupcima da ne plate više nego što objekat zaista vredi, dok prodavcima omogućavaju da postave realnu i konkurentnu cenu. Investitori koriste predviđanja cena kako bi odabrali najisplativije lokacije i trenutke za ulaganje, dok banke i finansijske institucije procenjuju rizik prilikom odobravanja kredita i hipoteka. Takođe, vlasti i urbanisti mogu na osnovu takvih podataka da planiraju razvoj gradova i prate trendove u stanogradnji. Na širem nivou, predviđanje cena kuća doprinosi stabilnosti tržišta nekretnina i sprečava formiranje ekonomskih balona koji mogu dovesti do finansijskih kriza.

Tradicionalno, procene vrednosti nekretnina često su se zasnivale na ličnom iskustvu agenata, subjektivnim procenama i ograničenom broju informacija sa tržišta. Takav pristup je bio spor, nepouzdan i podložan ljudskim greškama. Danas, zahvaljujući računarima i velikim količinama podataka, ovaj proces postaje daleko precizniji i objektivniji. Algoritmi i modeli analize podataka mogu automatski obrađivati hiljade parametara kao što su lokacija, površina, starost objekta, blizina infrastrukture i istorijski trendovi cena i na osnovu njih predvideti realnu vrednost kuće. Na taj način, savremene tehnološke metode zamenjuju subjektivne procene podacima zasnovanim na dokazima, čime se povećava tačnost, brzina i pouzdanost celog procesa procene vrednosti nekretnina.

Za potrebe ovog projekta korišćen je R programski jezik kao odličan alat jer nudi širok spektar biblioteka za statističku analizu, obradu podataka i vizualizaciju rezultata. Posebno je koristan za regresione analize i izgradnju prediktivnih modela, što je ključno za problem predviđanja cena kuća. Uz to, R omogućava lako prikazivanje i interpretaciju nalaza kroz grafove i tabele, čime se olakšava donošenje zaključaka iz podataka.

## Opis problema i motivacija

Iz ugla nauke o podacima, problem predviđanja cena kuća predstavlja zadatak regresije, gde se na osnovu poznatih karakteristika nekretnina (kao što su lokacija, površina, broj soba, starost i sl.) pokušava proceniti njihova tržišna vrednost. Cilj je izgraditi model koji može naučiti obrasce i odnose između tih karakteristika i cene, a zatim ih koristiti za predviđanje cena novih, nepoznatih kuća.

U ovom istraživanju koristimo javno dostupan Ames Housing Dataset koji sadrži podatke o više od 2.900 kuća prodatih u Amesu, Ajova, od 2006. do 2010. godine. Skup obuhvata preko 80 atributa koji opisuju različite karakteristike kuća, poput veličine, broja soba, kvaliteta materijala i tipa naselja. Ovaj skup se često koristi u istraživanjima i obuci modela mašinskog učenja jer pruža bogate i realistične podatke za analizu tržišta nekretnina.

Zadatak projekta je razviti model koji precizno predviđa prodajnu cenu kuće (SalePrice) na osnovu njenih karakteristika i uslova prodaje. Glavni izazov je konstruisanje robusnog i prediktivnog modela s obzirom na složenost podataka: 1) visoka dimenzionalnost zahteva efikasan izbor karakteristika da bi se izbeglo prekomerno prilagođavanje; 2) mešoviti tipovi podataka zahtevaju pažljivo kodiranje; i 3) potencijalna kolinearnost zahteva stabilizaciju.

Cilj ovog projekta je uporediti različite modele mašinskog učenja kako bi se identifikovao onaj koji najtačnije predviđa cene kuća. Evaluacijom performansi modela analizira se njihova sposobnost da uhvate odnose između karakteristika nekretnina i prodajne cene. Na taj način dolazi se do optimalnog pristupa za pouzdanu procenu tržišne vrednosti kuća.

## Opis i priprema podataka

### Izvor podataka

Za potrebe ovog istraživanja korišćen je Ames Housing Dataset, preuzet sa Kaggle platforme u okviru takmičenja House Prices: Advanced Regression Techniques. Skup sadrži detaljne informacije o 2919 stambenih objekata u gradu Ames, država Ajova, prikupljene između 2006. i 2010. godine. Podaci obuhvataju 1460 primera u trening skupu i 1459 primera u test skupu. U trening skupu, svaki primer je opisan pomoću 81 atributa, pri čemu je jedna promenljiva jedinstveni identifikator (Id), a jedna je ciljna promenljiva (SalePrice).

Nakon učitavanja podataka, funkcijom `str` proveravamo strukturu obeležja. Može se uočiti da ima ukupno 81 obeležje, od čega, 43 obeležja znakovnog tipa (`chr`) i 38 obeležja numeričkog tipa (`int`).

```

str(data)

## 'data.frame': 1460 obs. of  81 variables:
## $ Id      : int  1 2 3 4 5 6 7 8 9 10 ...
## $ MSSubClass : int  60 20 60 70 60 50 20 60 50 190 ...
## $ MSZoning  : chr "RL" "RL" "RL" "RL" ...
## $ LotFrontage: int  65 80 68 60 84 85 75 NA 51 50 ...
## $ LotArea   : int  8450 9600 11250 9550 14260 14115 10084 10382 6120 7
420 ...
## $ Street    : chr "Pave" "Pave" "Pave" "Pave" ...
## $ Alley     : chr NA NA NA NA ...
## $ LotShape   : chr "Reg" "Reg" "IR1" "IR1" ...
## $ LandContour: chr "Lvl" "Lvl" "Lvl" "Lvl" ...
## $ Utilities  : chr "AllPub" "AllPub" "AllPub" "AllPub" ...
## $ LotConfig  : chr "Inside" "FR2" "Inside" "Corner" ...
## $ LandSlope   : chr "Gtl" "Gtl" "Gtl" "Gtl" ...
## $ Neighborhood: chr "CollgCr" "Veenker" "CollgCr" "Crawfor" ...
## $ Condition1 : chr "Norm" "Feedr" "Norm" "Norm" ...
## $ Condition2 : chr "Norm" "Norm" "Norm" "Norm" ...
## $ BldgType   : chr "1Fam" "1Fam" "1Fam" "1Fam" ...
## $ HouseStyle  : chr "2Story" "1Story" "2Story" "2Story" ...
## $ OverallQual: int  7 6 7 7 8 5 8 7 7 5 ...
## $ OverallCond : int  5 8 5 5 5 5 5 6 5 6 ...
## $ YearBuilt   : int  2003 1976 2001 1915 2000 1993 2004 1973 1931 1939 .
..
## $ YearRemodAdd: int  2003 1976 2002 1970 2000 1995 2005 1973 1950 1950 .
..
## $ RoofStyle   : chr "Gable" "Gable" "Gable" "Gable" ...
## $ RoofMatl    : chr "CompShg" "CompShg" "CompShg" "CompShg" ...
## $ Exterior1st  : chr "VinylSd" "MetalSd" "VinylSd" "Wd Sdng" ...
## $ Exterior2nd  : chr "VinylSd" "MetalSd" "VinylSd" "Wd Shng" ...
## $ MasVnrType  : chr "BrkFace" "None" "BrkFace" "None" ...
## $ MasVnrArea  : int  196 0 162 0 350 0 186 240 0 0 ...
## $ ExterQual   : chr "Gd" "TA" "Gd" "TA" ...
## $ ExterCond   : chr "TA" "TA" "TA" "TA" ...
## $ Foundation  : chr "PConc" "CBlock" "PConc" "BrkTil" ...
## $ BsmtQual    : chr "Gd" "Gd" "Gd" "TA" ...
## $ BsmtCond    : chr "TA" "TA" "TA" "Gd" ...
## $ BsmtExposure: chr "No" "Gd" "Mn" "No" ...
## $ BsmtFinType1: chr "GLQ" "ALQ" "GLQ" "ALQ" ...
## $ BsmtFinSF1  : int  706 978 486 216 655 732 1369 859 0 851 ...
## $ BsmtFinType2: chr "Unf" "Unf" "Unf" "Unf" ...
## $ BsmtFinSF2  : int  0 0 0 0 0 32 0 0 ...
## $ BsmtUnfSF   : int  150 284 434 540 490 64 317 216 952 140 ...
## $ TotalBsmtSF: int  856 1262 920 756 1145 796 1686 1107 952 991 ...
## $ Heating     : chr "GasA" "GasA" "GasA" "GasA" ...
## $ HeatingQC   : chr "Ex" "Ex" "Ex" "Gd" ...
## $ CentralAir  : chr "Y" "Y" "Y" "Y" ...
## $ Electrical  : chr "SBrkr" "SBrkr" "SBrkr" "SBrkr" ...
## $ X1stFlrSF   : int  856 1262 920 961 1145 796 1694 1107 1022 1077 ...

```

```

## $ X2ndFlrSF : int 854 0 866 756 1053 566 0 983 752 0 ...
## $ LowQualFinSF : int 0 0 0 0 0 0 0 0 0 ...
## $ GrLivArea : int 1710 1262 1786 1717 2198 1362 1694 2090 1774 1077 .
..
## $ BsmtFullBath : int 1 0 1 1 1 1 1 1 0 1 ...
## $ BsmtHalfBath : int 0 1 0 0 0 0 0 0 0 ...
## $ FullBath : int 2 2 2 1 2 1 2 2 2 1 ...
## $ HalfBath : int 1 0 1 0 1 1 0 1 0 0 ...
## $ BedroomAbvGr : int 3 3 3 3 4 1 3 3 2 2 ...
## $ KitchenAbvGr : int 1 1 1 1 1 1 1 1 2 2 ...
## $ KitchenQual : chr "Gd" "TA" "Gd" "Gd" ...
## $ TotRmsAbvGrd : int 8 6 6 7 9 5 7 7 8 5 ...
## $ Functional : chr "Typ" "Typ" "Typ" "Typ" ...
## $ Fireplaces : int 0 1 1 1 1 0 1 2 2 2 ...
## $ FireplaceQu : chr NA "TA" "TA" "Gd" ...
## $ GarageType : chr "Attchd" "Attchd" "Attchd" "Detchd" ...
## $ GarageYrBlt : int 2003 1976 2001 1998 2000 1993 2004 1973 1931 1939 .
..
## $ GarageFinish : chr "RFn" "RFn" "RFn" "Unf" ...
## $ GarageCars : int 2 2 2 3 3 2 2 2 2 1 ...
## $ GarageArea : int 548 460 608 642 836 480 636 484 468 205 ...
## $ GarageQual : chr "TA" "TA" "TA" "TA" ...
## $ GarageCond : chr "TA" "TA" "TA" "TA" ...
## $ PavedDrive : chr "Y" "Y" "Y" "Y" ...
## $ WoodDeckSF : int 0 298 0 0 192 40 255 235 90 0 ...
## $ OpenPorchSF : int 61 0 42 35 84 30 57 204 0 4 ...
## $ EnclosedPorch: int 0 0 0 272 0 0 0 228 205 0 ...
## $ X3SsnPorch : int 0 0 0 0 0 320 0 0 0 0 ...
## $ ScreenPorch : int 0 0 0 0 0 0 0 0 0 0 ...
## $ PoolArea : int 0 0 0 0 0 0 0 0 0 0 ...
## $ PoolQC : chr NA NA NA NA ...
## $ Fence : chr NA NA NA NA ...
## $ MiscFeature : chr NA NA NA NA ...
## $ MiscVal : int 0 0 0 0 0 700 0 350 0 0 ...
## $ MoSold : int 2 5 9 2 12 10 8 11 4 1 ...
## $ YrSold : int 2008 2007 2008 2006 2008 2009 2007 2009 2008 2008 .
..
## $ SaleType : chr "WD" "WD" "WD" "WD" ...
## $ SaleCondition: chr "Normal" "Normal" "Normal" "Abnorml" ...
## $ SalePrice : int 208500 181500 223500 140000 250000 143000 307000 20
000 129900 118000 ...

```

U nastavku se nalazi spisak obeležja i kratak opis za svako:

1. **Id** – Jedinstveni identifikator.
2. **MSSubClass** – Klasa građevine.
3. **MSZoning** – Opšta zonacija zemljišta.
4. **LotFrontage** – Dužina placa povezanog sa ulicom (u stopama).
5. **LotArea** – Površina placa u kvadratnim stopama.
6. **Street** – Tip pristupnog puta.
7. **Alley** – Tip pristupa prolazom.
8. **LotShape** – Opšti oblik parcele.
9. **LandContour** – Ravnost zemljišta.
10. **Utilities** – Tip dostupnih komunalnih usluga.
11. **LotConfig** – Konfiguracija parcele.
12. **LandSlope** – Nagib zemljišta.
13. **Neighborhood** – Fizička lokacija unutar grada Ames.
14. **Condition1** – Blizina glavnog puta ili železnice.
15. **Condition2** – Blizina glavnog puta ili železnice (ako postoji drugi).
16. **BldgType** – Tip objekta.
17. **HouseStyle** – Stil kuće.
18. **OverallQual** – Ukupan kvalitet materijala i završne obrade.
19. **OverallCond** – Ukupno stanje objekta.
20. **YearBuilt** – Godina izgradnje.
21. **YearRemodAdd** – Godina renoviranja ili dogradnje.
22. **RoofStyle** – Tip krova.
23. **RoofMatl** – Materijal krova.
24. **Exterior1st** – Spoljašnja obloga kuće.
25. **Exterior2nd** – Drugi tip spoljašnje obloge (ako postoji).
26. **MasVnrType** – Tip malterisanja ili dekorativnog kamena.

27. **MasVnrArea** – Površina malterisanja ili dekorativnog kamenja u kvadratnim stopama.
28. **ExterQual** – Kvalitet spoljašnjeg materijala.
29. **ExterCond** – Trenutno stanje spoljašnjeg materijala.
30. **Foundation** – Tip temelja.
31. **BsmtQual** – Visina podruma.
32. **BsmtCond** – Opšte stanje podruma.
33. **BsmtExposure** – Podrum sa izlazom ili na nivou bašte.
34. **BsmtFinType1** – Kvalitet završene površine podruma (prvi tip).
35. **BsmtFinSF1** – Površina završenog prostora u kvadratnim stopama (prvi tip).
36. **BsmtFinType2** – Kvalitet završene površine podruma (drugi tip, ako postoji).
37. **BsmtFinSF2** – Površina završenog prostora u kvadratnim stopama (drugi tip).
38. **BsmtUnfSF** – Nepotpuno završeni prostor podruma u kvadratnim stopama.
39. **TotalBsmtSF** – Ukupna površina podruma u kvadratnim stopama.
40. **Heating** – Tip grejanja.
41. **HeatingQC** – Kvalitet i stanje grejanja.
42. **CentralAir** – Centralno klimatizovanje (da/ne).
43. **Electrical** – Elektroinstalacije.
44. **1stFlrSF** – Površina prvog sprata u kvadratnim stopama.
45. **2ndFlrSF** – Površina drugog sprata u kvadratnim stopama.
46. **LowQualFinSF** – Kvadratura završene površine niskog kvaliteta.
47. **GrLivArea** – Površina stambenog prostora iznad nivoa zemlje.
48. **BsmtFullBath** – Broj punih kupatila u podrumu.
49. **BsmtHalfBath** – Broj polu-kupatila u podrumu.
50. **FullBath** – Broj punih kupatila iznad nivoa zemlje.
51. **HalfBath** – Broj polu-kupatila iznad nivoa zemlje.
52. **Bedroom** – Broj spavačih soba iznad podruma.

53. **Kitchen** – Broj kuhinja.
54. **KitchenQual** – Kvalitet kuhinje.
55. **TotRmsAbvGrd** – Ukupan broj soba iznad nivoa zemlje (bez kupatila).
56. **Functional** – Ocena funkcionalnosti kuće.
57. **Fireplaces** – Broj kamina.
58. **FireplaceQu** – Kvalitet kamina.
59. **GarageType** – Lokacija garaže.
60. **GarageYrBlt** – Godina izgradnje garaže.
61. **GarageFinish** – Završna obrada unutrašnjosti garaže.
62. **GarageCars** – Kapacitet garaže (broj automobila).
63. **GarageArea** – Površina garaže u kvadratnim stopama.
64. **GarageQual** – Kvalitet garaže.
65. **GarageCond** – Stanje garaže.
66. **PavedDrive** – Da li je prilaz asfaltiran.
67. **WoodDeckSF** – Površina drvene terase u kvadratnim stopama.
68. **OpenPorchSF** – Površina otvorene verande u kvadratnim stopama.
69. **EnclosedPorch** – Površina zatvorene verande u kvadratnim stopama.
70. **3SsnPorch** – Površina trisezonske verande u kvadratnim stopama.
71. **ScreenPorch** – Površina mrežaste verande u kvadratnim stopama.
72. **PoolArea** – Površina bazena u kvadratnim stopama.
73. **PoolQC** – Kvalitet bazena.
74. **Fence** – Kvalitet ograde.
75. **MiscFeature** – Ostale karakteristike koje nisu pokrivene drugim kategorijama.
76. **MiscVal** – Vrednost dodatne karakteristike u dolarima.
77. **MoSold** – Mesec prodaje.
78. **YrSold** – Godina prodaje.
79. **SaleType** – Tip prodaje.

80. **SaleCondition** – Uslovi prodaje.

81. **SalePrice** – Prodajna cena objekta u dolarima (ciljna promenljiva).

Ciljna promenljiva, SalePrice, predstavlja prodajnu cenu kuće u američkim dolarima. Njena distribucija je ključna za modelovanje:

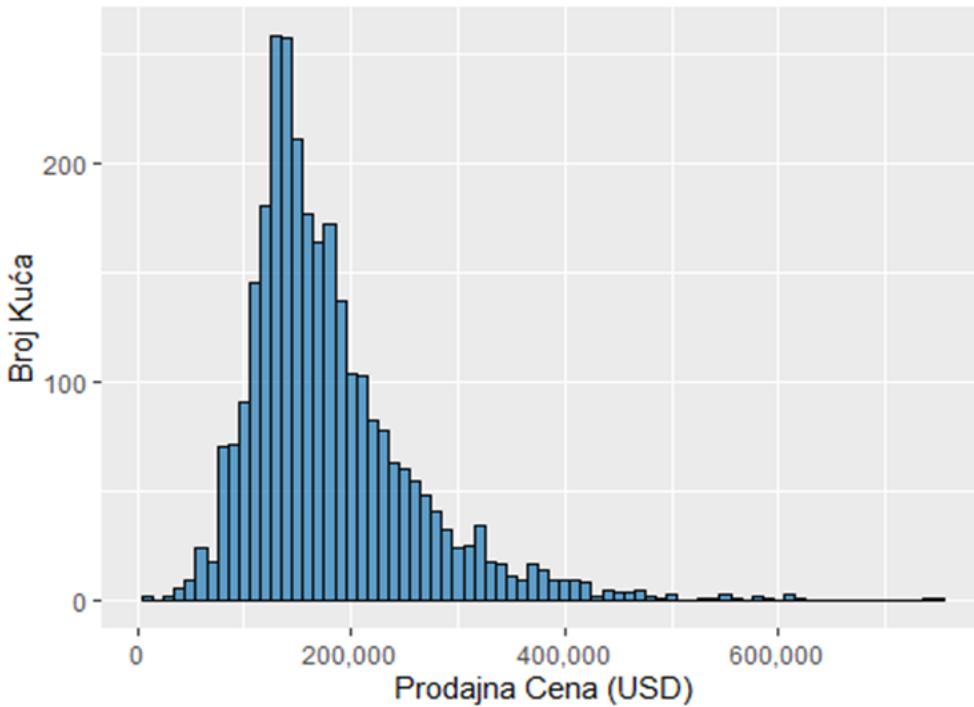
```
summary(data$SalePrice)

##   Min. 1st Qu. Median   Mean 3rd Qu.   Max.
## 34900 129975 163000 180921 214000 755000
```

Srednja vrednost (mean) je veća od medijane. Postoji nekoliko vrlo skupih kuća koje povlače srednju vrednost iznad medijane. Većina kuća je grupisana oko nižih i srednjih cena, a mali broj dostiže ekstremno visoke cene.

```
ggplot(data, aes(x = SalePrice)) +
  geom_histogram(binwidth = 10000, fill = "#1f78b4", color = "black", alpha = 0.7) +
  scale_x_continuous(labels = comma) +
  labs(
    title = "Distribucija Prodajne Cene",
    x = "Prodajna Cena (USD)",
    y = "Broj Kuća"
  ) +
  theme(plot.title = element_text(hjust = 0.5, face = "bold"))
```

## Distribucija Prodajne Cene



Linearna regresija se bazira na pretpostavci da su reziduali normalno distribuirani. Kada je ciljna promenljiva asimetrična, model će biti pristrasan prema podacima koji imaju ekstremne vrednosti (outlieri), jer oni izazivaju najveće greške (reziduale) i model će pokušati da se prilagodi njima. Rešenje za ovaj problem je logaritamska transformacija koja smanjuje uticaj ekstremnih vrednosti. Distribucija postaje bliža normalnoj, model je stabilniji i precizniji, a predikcije manje osetljive na outliere.

## Čišćenje i obrada podataka

Uklanjanje outliera je važan korak u obradi podataka jer ekstremne vrednosti mogu značajno iskriviti rezultate analize i negativno uticati na performanse modela. Cilj ovog postupka je poboljšati tačnost i stabilnost modela uklanjanjem posmatranja koja se ne uklapaju u opšti obrazac podataka.

Za identifikaciju outliera korišćeni su scatter plotovi koji prikazuju odnos između numeričkih prediktora i ciljne promenljive SalePrice. Na osnovu vizuelne inspekcije izdvojeni su redovi sa neuobičajeno visokim ili niskim vrednostima, nakon čega su njihovi indeksi pronađeni pomoću funkcije filter() i ti redovi su uklonjeni iz skupa podataka.

Koristili smo funkciju `str()` kako bismo proverili strukturu i tipove podataka u skupu i identifikovali koje promenljive imaju numeričke vrednosti. Poseban fokus bio je na numeričkim promenljivama, jer su one ključne za vizuelizaciju odnosa sa ciljnom promenljivom `SalePrice` putem scatter plotova. Ova provera nam je omogućila da izdvojimo odgovarajuće podatke za analizu i detekciju outliera.

```
str(data)
```

```
## 'data.frame': 1460 obs. of  81 variables:
##   $ Id           : int  1 2 3 4 5 6 7 8 9 10 ...
##   $ MSSubClass  : int  60 20 60 70 60 50 20 60 50 190 ...
##   $ MSZoning    : chr  "RL" "RL" "RL" "RL" ...
##   $ LotFrontage : int  65 80 68 60 84 85 75 NA 51 50 ...
##   $ LotArea     : int  8450 9600 11250 9550 14260 14115 10084 10382 6120
7420 ...
##   $ Street       : chr  "Pave" "Pave" "Pave" "Pave" ...
##   $ Alley        : chr  NA NA NA NA ...
##   $ LotShape     : chr  "Reg" "Reg" "IR1" "IR1" ...
##   $ LandContour  : chr  "Lvl" "Lvl" "Lvl" "Lvl" ...
##   $ Utilities    : chr  "AllPub" "AllPub" "AllPub" "AllPub" ...
##   $ LotConfig    : chr  "Inside" "FR2" "Inside" "Corner" ...
##   $ LandSlope    : chr  "Gtl" "Gtl" "Gtl" "Gtl" ...
##   $ Neighborhood : chr  "CollgCr" "Veenker" "CollgCr" "Crawfor" ...
##   $ Condition1   : chr  "Norm" "Feedr" "Norm" "Norm" ...
##   $ Condition2   : chr  "Norm" "Norm" "Norm" "Norm" ...
##   $ BldgType     : chr  "1Fam" "1Fam" "1Fam" "1Fam" ...
##   $ HouseStyle   : chr  "2Story" "1Story" "2Story" "2Story" ...
##   $ OverallQual  : int  7 6 7 7 8 5 8 7 7 5 ...
##   $ OverallCond  : int  5 8 5 5 5 5 5 6 5 6 ...
##   $ YearBuilt    : int  2003 1976 2001 1915 2000 1993 2004 1973 1931
1939 ...
##   $ YearRemodAdd : int  2003 1976 2002 1970 2000 1995 2005 1973 1950
1950 ...
##   $ RoofStyle    : chr  "Gable" "Gable" "Gable" "Gable" ...
##   $ RoofMatl    : chr  "CompShg" "CompShg" "CompShg" "CompShg" ...
##   $ Exterior1st  : chr  "VinylSd" "MetalSd" "VinylSd" "Wd Sdng" ...
##   $ Exterior2nd  : chr  "VinylSd" "MetalSd" "VinylSd" "Wd Shng" ...
##   $ MasVnrType   : chr  "BrkFace" "None" "BrkFace" "None" ...
##   $ MasVnrArea   : int  196 0 162 0 350 0 186 240 0 0 ...
##   $ ExterQual    : chr  "Gd" "TA" "Gd" "TA" ...
##   $ ExterCond    : chr  "TA" "TA" "TA" "TA" ...
##   $ Foundation   : chr  "PConc" "CBlock" "PConc" "BrkTil" ...
```

```

## $ BsmtQual      : chr  "Gd" "Gd" "Gd" "TA" ...
## $ BsmtCond      : chr  "TA" "TA" "TA" "Gd" ...
## $ BsmtExposure  : chr  "No" "Gd" "Mn" "No" ...
## $ BsmtFinType1  : chr  "GLQ" "ALQ" "GLQ" "ALQ" ...
## $ BsmtFinSF1    : int   706 978 486 216 655 732 1369 859 0 851 ...
## $ BsmtFinType2  : chr  "Unf" "Unf" "Unf" "Unf" ...
## $ BsmtFinSF2    : int   0 0 0 0 0 0 32 0 0 ...
## $ BsmtUnfSF    : int   150 284 434 540 490 64 317 216 952 140 ...
## $ TotalBsmtSF   : int   856 1262 920 756 1145 796 1686 1107 952 991 ...
## $ Heating        : chr  "GasA" "GasA" "GasA" "GasA" ...
## $ HeatingQC     : chr  "Ex" "Ex" "Ex" "Gd" ...
## $ CentralAir    : chr  "Y" "Y" "Y" "Y" ...
## $ Electrical    : chr  "SBrkr" "SBrkr" "SBrkr" "SBrkr" ...
## $ X1stFlrSF    : int   856 1262 920 961 1145 796 1694 1107 1022 1077 ...
## $ X2ndFlrSF    : int   854 0 866 756 1053 566 0 983 752 0 ...
## $ LowQualFinSF  : int   0 0 0 0 0 0 0 0 0 ...
## $ GrLivArea     : int   1710 1262 1786 1717 2198 1362 1694 2090 1774
1077 ...
## $ BsmtFullBath  : int   1 0 1 1 1 1 1 1 0 1 ...
## $ BsmtHalfBath  : int   0 1 0 0 0 0 0 0 0 ...
## $ FullBath      : int   2 2 2 1 2 1 2 2 2 1 ...
## $ HalfBath      : int   1 0 1 0 1 1 0 1 0 0 ...
## $ BedroomAbvGr  : int   3 3 3 3 4 1 3 3 2 2 ...
## $ KitchenAbvGr  : int   1 1 1 1 1 1 1 1 2 2 ...
## $ KitchenQual    : chr  "Gd" "TA" "Gd" "Gd" ...
## $ TotRmsAbvGrd  : int   8 6 6 7 9 5 7 7 8 5 ...
## $ Functional    : chr  "Typ" "Typ" "Typ" "Typ" ...
## $ Fireplaces    : int   0 1 1 1 1 0 1 2 2 2 ...
## $ FireplaceQu   : chr  NA "TA" "TA" "Gd" ...
## $ GarageType    : chr  "Attchd" "Attchd" "Attchd" "Detchd" ...
## $ GarageYrBlt   : int   2003 1976 2001 1998 2000 1993 2004 1973 1931
1939 ...
## $ GarageFinish   : chr  "RFn" "RFn" "RFn" "Unf" ...
## $ GarageCars    : int   2 2 2 3 3 2 2 2 2 1 ...
## $ GarageArea    : int   548 460 608 642 836 480 636 484 468 205 ...
## $ GarageQual    : chr  "TA" "TA" "TA" "TA" ...
## $ GarageCond    : chr  "TA" "TA" "TA" "TA" ...
## $ PavedDrive    : chr  "Y" "Y" "Y" "Y" ...
## $ WoodDeckSF    : int   0 298 0 0 192 40 255 235 90 0 ...
## $ OpenPorchSF   : int   61 0 42 35 84 30 57 204 0 4 ...
## $ EnclosedPorch : int   0 0 0 272 0 0 0 228 205 0 ...
## $ X3SsnPorch   : int   0 0 0 0 0 320 0 0 0 0 ...
## $ ScreenPorch   : int   0 0 0 0 0 0 0 0 0 0 ...

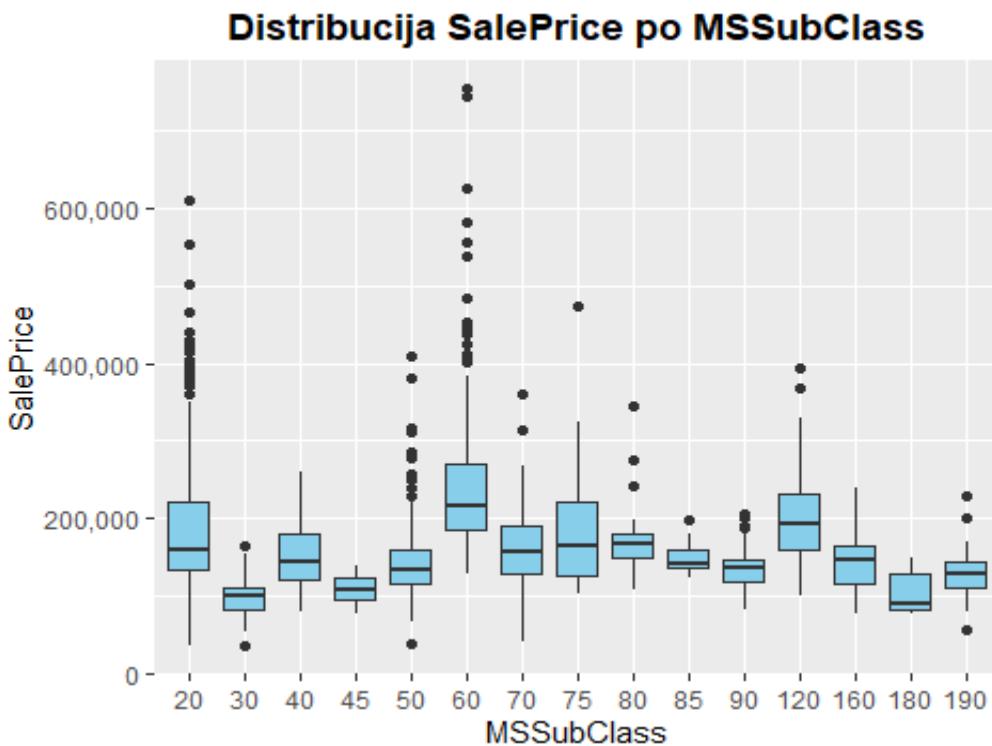
```

```

## $ PoolArea      : int  0 0 0 0 0 0 0 0 0 0 ...
## $ PoolQC       : chr  NA NA NA NA ...
## $ Fence        : chr  NA NA NA NA ...
## $ MiscFeature   : chr  NA NA NA NA ...
## $ MiscVal       : int  0 0 0 0 700 0 350 0 0 ...
## $ MoSold        : int  2 5 9 2 12 10 8 11 4 1 ...
## $ YrSold        : int  2008 2007 2008 2006 2008 2009 2007 2009 2008
2008 ...
## $ SaleType       : chr  "WD" "WD" "WD" "WD" ...
## $ SaleCondition: chr  "Normal" "Normal" "Normal" "Abnorml" ...
## $ SalePrice      : int  208500 181500 223500 140000 250000 143000 307000
200000 129900 118000 ...

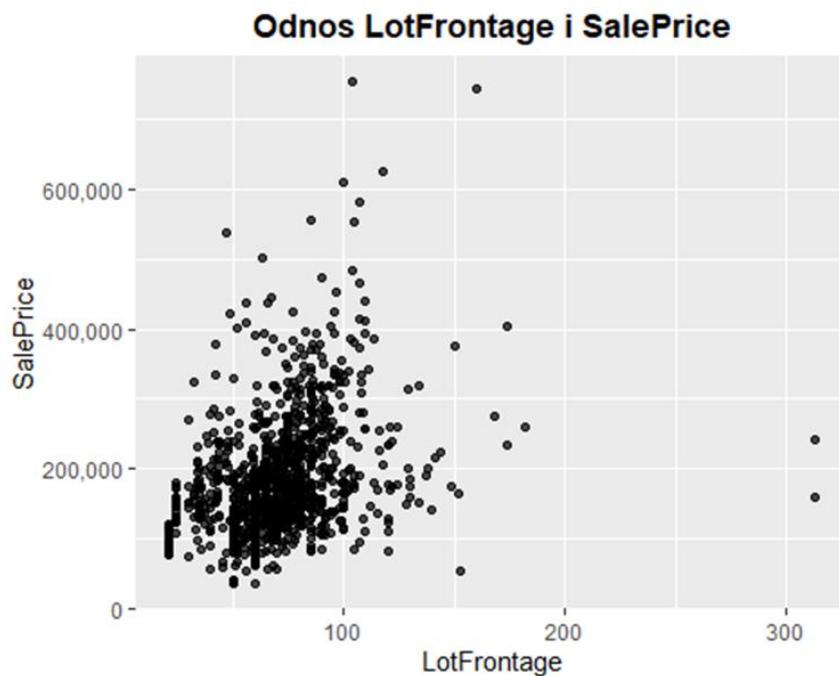
ggplot(data.train, aes(x = as.factor(MSSubClass), y = SalePrice)) +
  geom_boxplot(fill = "skyblue") +
  scale_y_continuous(labels = comma) +
  labs(
    title = "Distribucija SalePrice po MSSubClass",
    x = "MSSubClass",
    y = "SalePrice"
  ) +
  theme(plot.title = element_text(hjust = 0.5, face = "bold"))

```



Grafik prikazuje raspodelu cena kuća po različitim tipovima. Nije uočena nijedna vrednost koja značajnije odstupa od ostatka podataka.

```
ggplot(data.train, aes(x = LotFrontage, y = SalePrice)) +  
  geom_point(alpha = 0.7) +  
  scale_x_continuous(labels = comma) +  
  scale_y_continuous(labels = comma) +  
  labs(  
    title = "Odnos LotFrontage i SalePrice",  
    x = "LotFrontage",  
    y = "SalePrice"  
) +  
  theme(plot.title = element_text(hjust = 0.5, face = "bold"))
```



Prilikom analize zavisnosti između LotFrontage i ciljne promenljive, uočene su dve vrednosti sa visokom polugom koje značajno odstupaju od ostalih podataka. Uklonićemo ih.

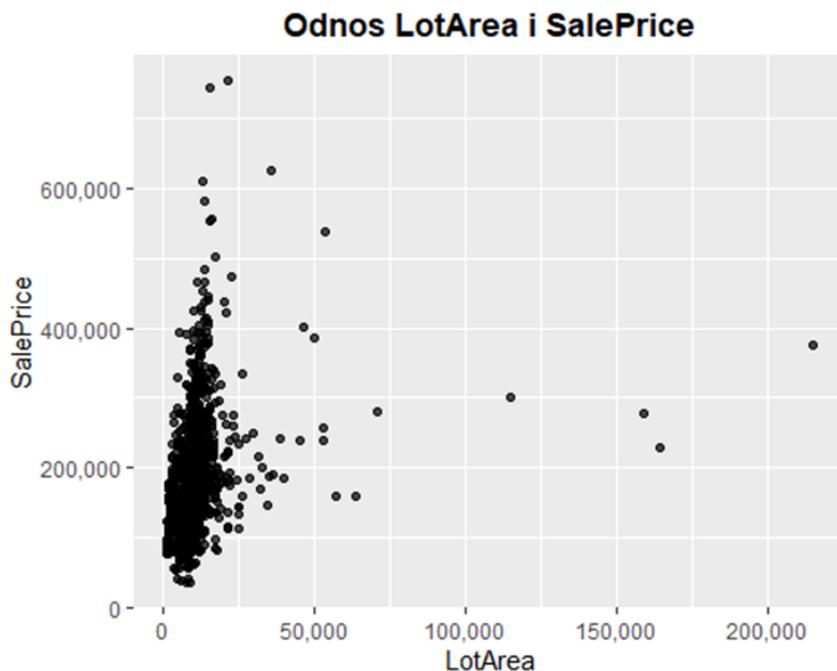
```

data.train %>% filter(LotFrontage > 300) %>% select(Order)

##      Order
## 1 1499
## 2 1266

ggplot(data.train, aes(x = LotArea, y = SalePrice)) +
  geom_point(alpha = 0.7) +
  scale_x_continuous(labels = comma) +
  scale_y_continuous(labels = comma) +
  labs(
    title = "Odnos LotArea i SalePrice",
    x = "LotArea",
    y = "SalePrice"
  ) +
  theme(plot.title = element_text(hjust = 0.5, face = "bold"))

```



Prilikom analize zavisnosti između LotArea i ciljne promenljive, uočene su vrednosti koje odaskaču od ostatka podataka. Te vrednosti ćemo ukloniti.

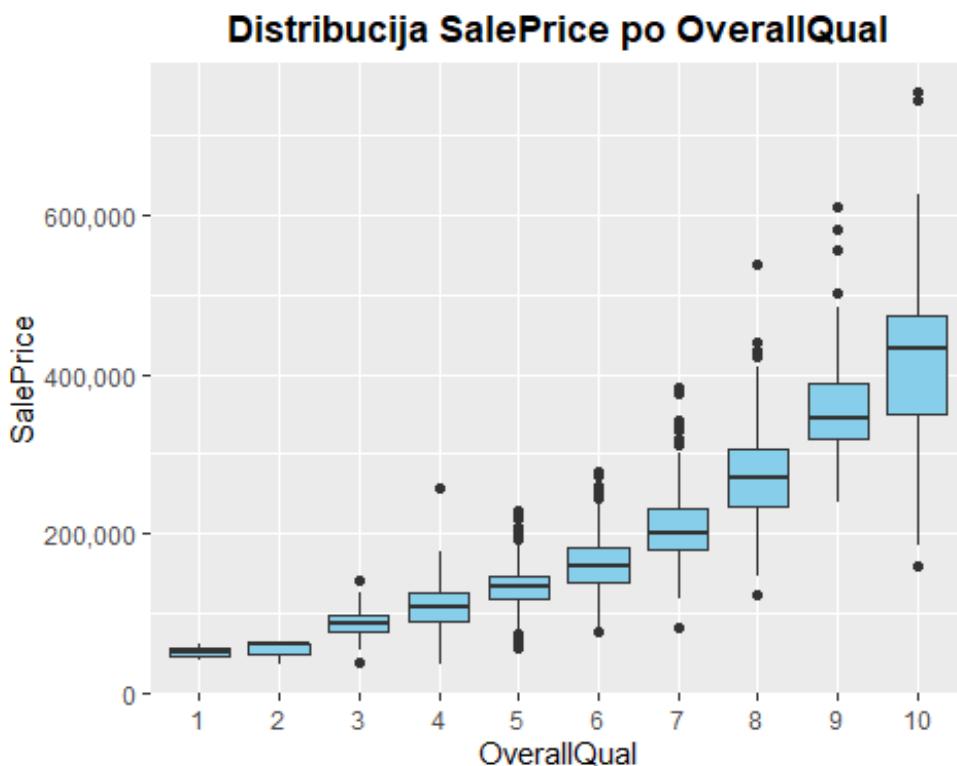
```

data.train %>% filter(LotArea > 55000) %>% select(Order)

##      Order
## 1 957
## 2 1571
## 3 2072
## 4 2116

ggplot(data.train, aes(x = as.factor(OverallQual), y = SalePrice)) +
  geom_boxplot(fill = "skyblue") +
  scale_y_continuous(labels = comma) +
  labs(
    title = "Distribucija SalePrice po OverallQual",
    x = "OverallQual",
    y = "SalePrice"
  ) +
  theme(plot.title = element_text(hjust = 0.5, face = "bold"))

```

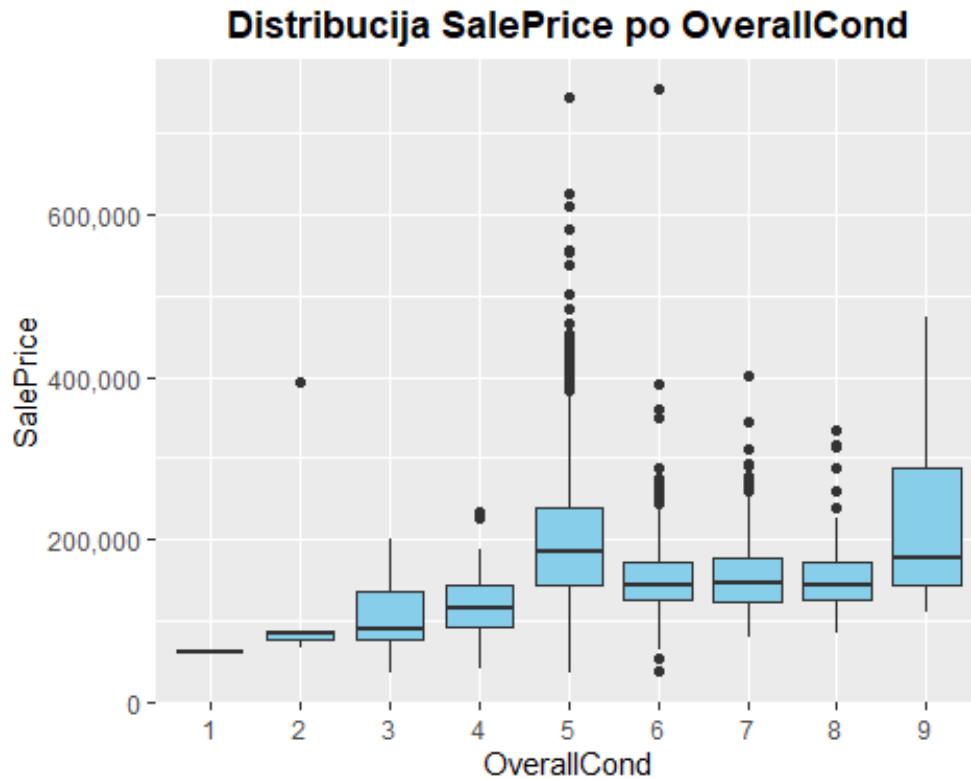


Na grafiku je predstavljena zavisnost cene kuće od ukupnog kvaliteta. Za OverallQual = 10, primetili smo dve kuće čija je cena značajno niža od očekivane. Da ne bi uticale na model, uklonićemo ih iz skupa podataka.

```
data.train %>% filter(OverallQual == 10 & SalePrice < 200000) %>%
  select(-Order)

##      Order
## 1    1499
## 2    2181

ggplot(data.train, aes(x = as.factor(OverallCond), y = SalePrice)) +
  geom_boxplot(fill = "skyblue") +
  scale_y_continuous(labels = comma) +
  labs(
    title = "Distribucija SalePrice po OverallCond",
    x = "OverallCond",
    y = "SalePrice"
  ) +
  theme(plot.title = element_text(hjust = 0.5, face = "bold"))
```



Analizirajući grafik zavisnosti cene kuće i njenog stanja, primetili smo nekoliko kuća čije cene značajno odskaču od očekivanih vrednosti:

- OverallCond = 6 - SalePrice > 600.000 (Id 692)
- OverallCond = 2 - SalePrice > 300.000 (Id 379)
- OverallCond = 5 - SalePrice > 700.000 (Id 1183)

Uklanjamo ih iz skupa.

```
data.train %>% filter(OverallCond == 6 & SalePrice > 600000) %>%
  select(Order)

##      Order
## 1 1768

data.train %>% filter(OverallCond == 2 & SalePrice > 300000) %>%
  select(Order)

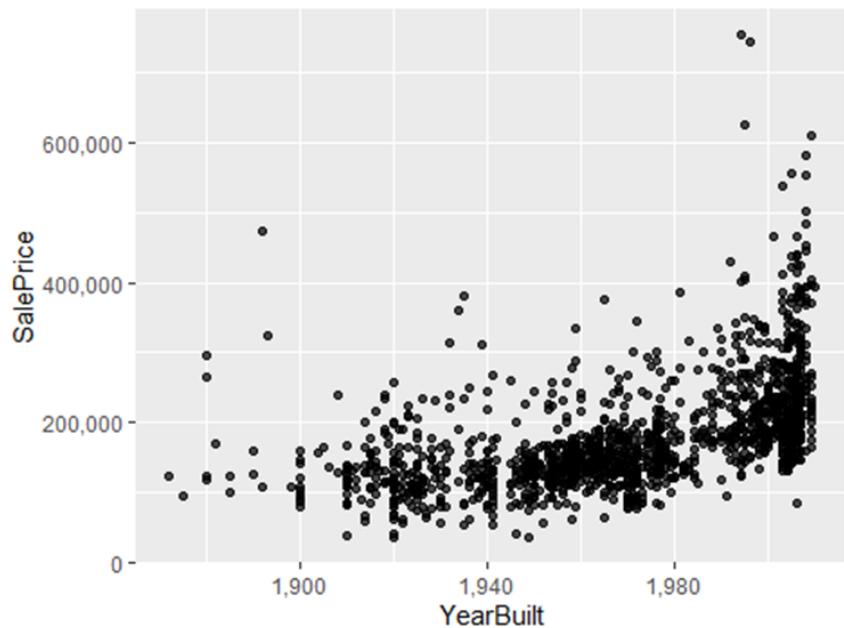
##      Order
## 1 18

data.train %>% filter(OverallCond == 5 & SalePrice > 700000) %>%
  select(Order)

##      Order
## 1 1761

ggplot(data.train, aes(x = YearBuilt, y = SalePrice)) +
  geom_point(alpha = 0.7) +
  scale_x_continuous(labels = comma) +
  scale_y_continuous(labels = comma) +
  labs(
    title = "Odnos YearBuilt i SalePrice",
    x = "YearBuilt",
    y = "SalePrice"
  ) +
  theme(plot.title = element_text(hjust = 0.5, face = "bold"))
```

Odnos YearBuilt i SalePrice



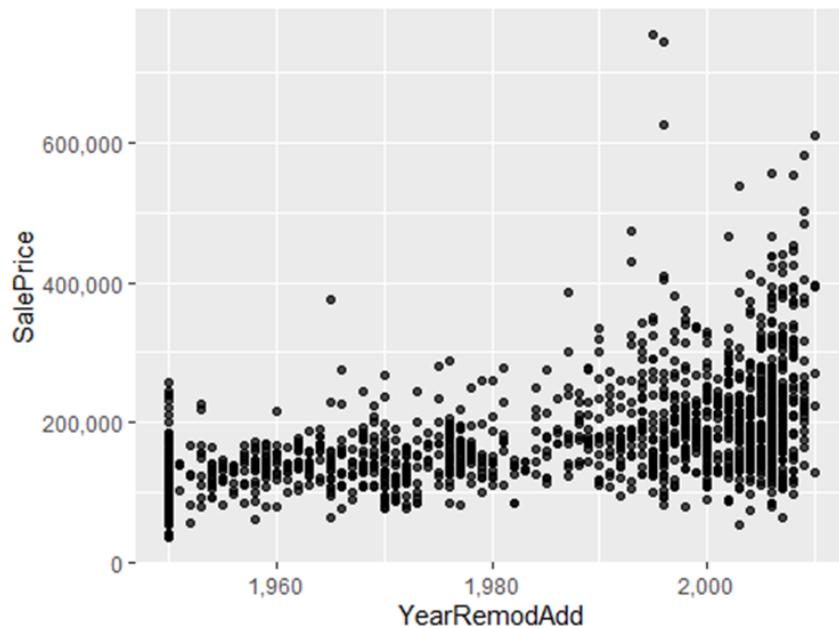
Uočena je kuća koja je izgrađena pre 1900. godine, a čija je cena znatno viša od očekivane. Ova vrednost se izdvojila od ostatka skupa podataka i biće uklonjena.

```
data.train %>% filter(YearBuilt < 1900 & SalePrice > 400000) %>%
  select(Order)

##      Order
## 1 2667

ggplot(data.train, aes(x = YearRemodAdd, y = SalePrice)) +
  geom_point(alpha = 0.7) +
  scale_x_continuous(labels = comma) +
  scale_y_continuous(labels = comma) +
  labs(
    title = "Odnos YearRemodAdd i SalePrice",
    x = "YearRemodAdd",
    y = "SalePrice"
  ) +
  theme(plot.title = element_text(hjust = 0.5, face = "bold"))
```

### Odnos YearRemodAdd i SalePrice



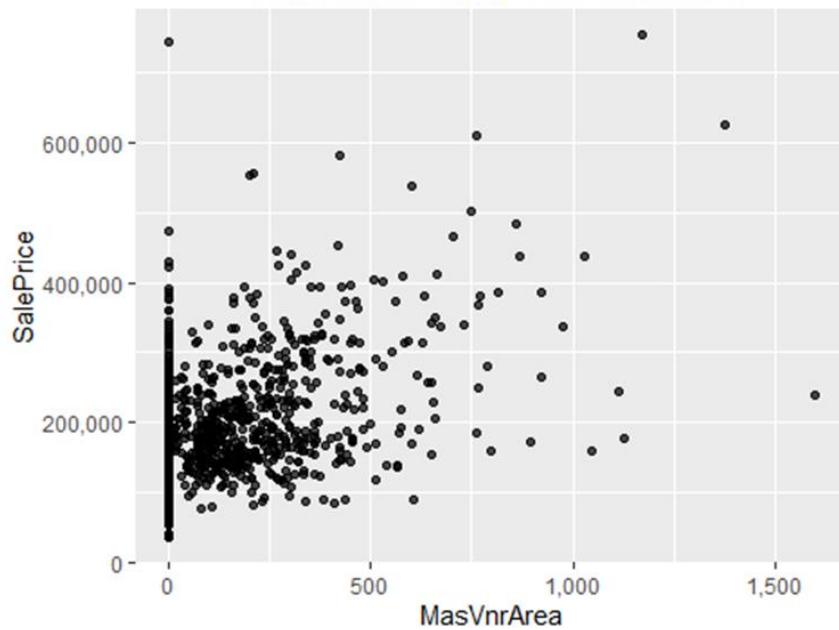
Primećena je kuća čija je godina poslednjeg renoviranja pre 1970, a cena je znatno viša od očekivane. Ova vrednost će biti uklonjena.

```
data.train %>% filter(YearRemodAdd < 1970 & SalePrice > 300000) %>%
  select(Order)

##      Order
## 1 957

ggplot(data.train, aes(x = MasVnrArea, y = SalePrice)) +
  geom_point(alpha = 0.7) +
  scale_x_continuous(labels = comma) +
  scale_y_continuous(labels = comma) +
  labs(
    title = "Odnos MasVnrArea i SalePrice",
    x = "MasVnrArea",
    y = "SalePrice"
  ) +
  theme(plot.title = element_text(hjust = 0.5, face = "bold"))
```

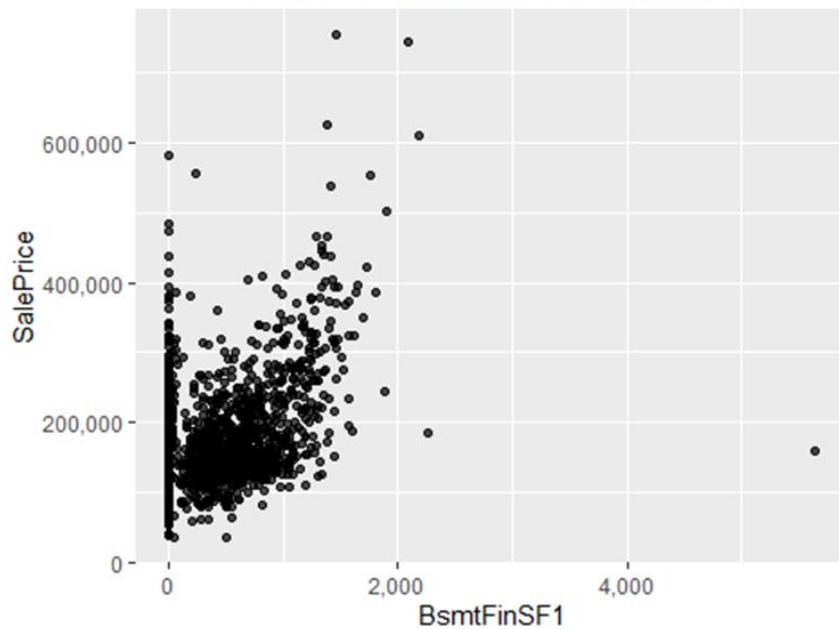
**Odnos MasVnrArea i SalePrice**



Na grafiku se može videti odnos između MasVnrArea i ciljne promenljive.

```
ggplot(data.train, aes(x = BsmtFinSF1, y = SalePrice)) +
  geom_point(alpha = 0.7) +
  scale_x_continuous(labels = comma) +
  scale_y_continuous(labels = comma) +
  labs(
    title = "Odnos BsmtFinSF1 i SalePrice",
    x = "BsmtFinSF1",
    y = "SalePrice"
  ) +
  theme(plot.title = element_text(hjust = 0.5, face = "bold"))
```

**Odnos BsmtFinSF1 i SalePrice**



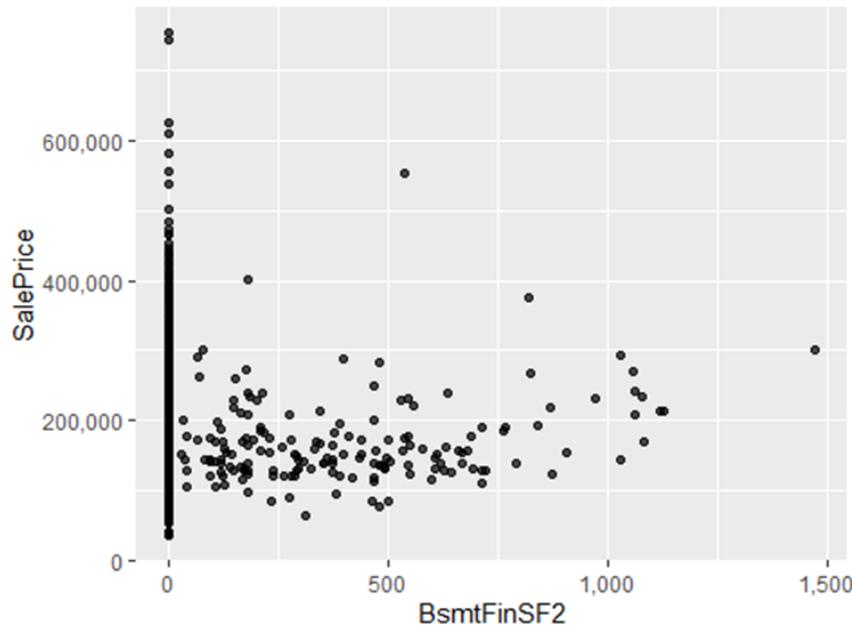
Prilikom analize površine završenog podruma, uočena je vrednost sa visokom polugom ( $BsmtFinSF1 > 4000$ ). Vrednost će biti uklonjena.

```
data.train %>% filter(BsmtFinSF1 > 4000) %>% select(-Order)

##      Order
## 1 1499

ggplot(data.train, aes(x = BsmtFinSF2, y = SalePrice)) +
  geom_point(alpha = 0.7) +
  scale_x_continuous(labels = comma) +
  scale_y_continuous(labels = comma) +
  labs(
    title = "Odnos BsmtFinSF2 i SalePrice",
    x = "BsmtFinSF2",
    y = "SalePrice"
  ) +
  theme(plot.title = element_text(hjust = 0.5, face = "bold"))
```

**Odnos BsmtFinSF2 i SalePrice**



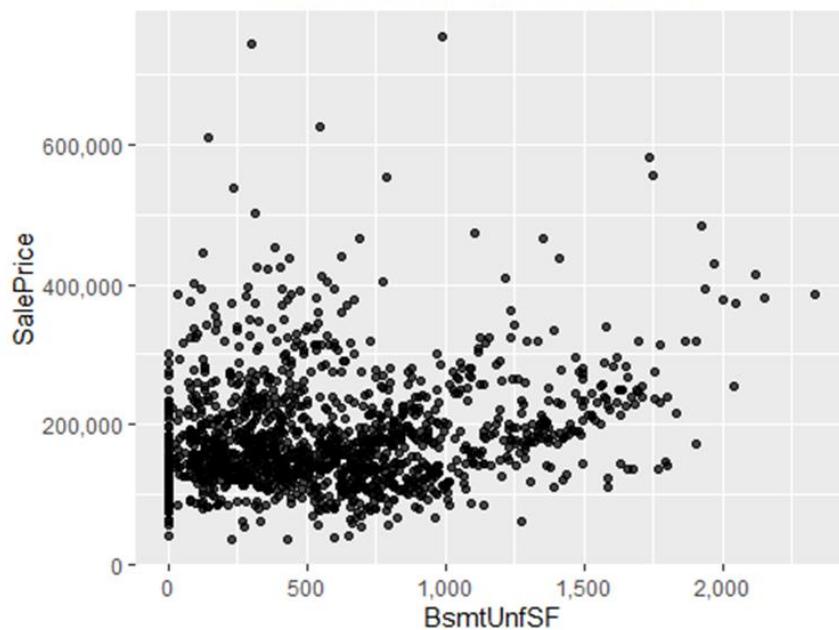
Prilikom analize druge površine završenog podruma, uočena je jedna izuzetno velika vrednost koja značajno odstupa od ostalih. Vrednost će biti uklonjena.

```
data.train %>% filter(BsmtFinSF2 > 500 & SalePrice > 500000) %>%
  select(Order)

##      Order
## 1 424

ggplot(data.train, aes(x = BsmtUnfSF, y = SalePrice)) +
  geom_point(alpha = 0.7) +
  scale_x_continuous(labels = comma) +
  scale_y_continuous(labels = comma) +
  labs(
    title = "Odnos BsmtUnfSF i SalePrice",
    x = "BsmtUnfSF",
    y = "SalePrice"
  ) +
  theme(plot.title = element_text(hjust = 0.5, face = "bold"))
```

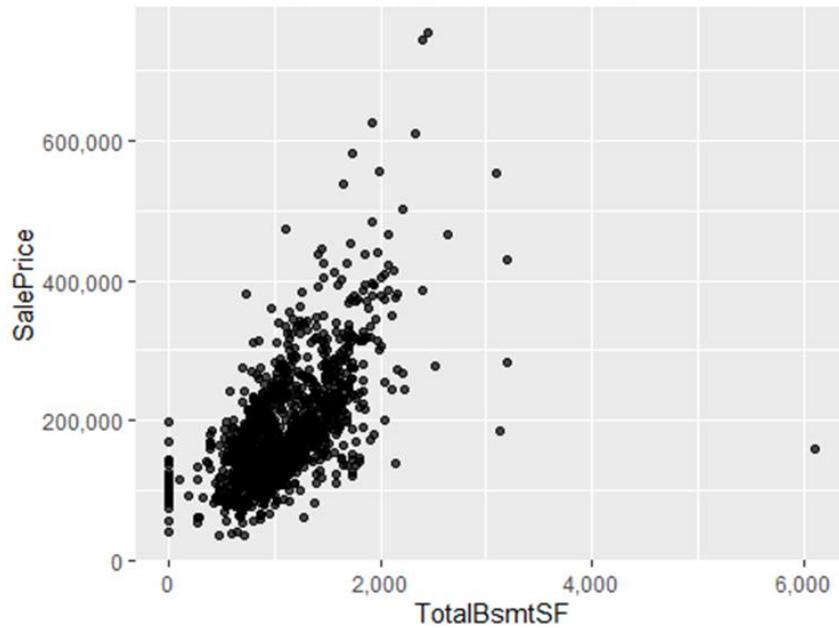
Odnos BsmtUnfSF i SalePrice



Grafikom je predstavljen odnos između ciljne promenljive i nezavršene površine u podrumu. Nisu uočeni outlieri.

```
ggplot(data.train, aes(x = TotalBsmtSF, y = SalePrice)) +  
  geom_point(alpha = 0.7) +  
  scale_x_continuous(labels = comma) +  
  scale_y_continuous(labels = comma) +  
  labs(  
    title = "Odnos TotalBsmtSF i SalePrice",  
    x = "TotalBsmtSF",  
    y = "SalePrice"  
  ) +  
  theme(plot.title = element_text(hjust = 0.5, face = "bold"))
```

### Odnos TotalBsmtSF i SalePrice



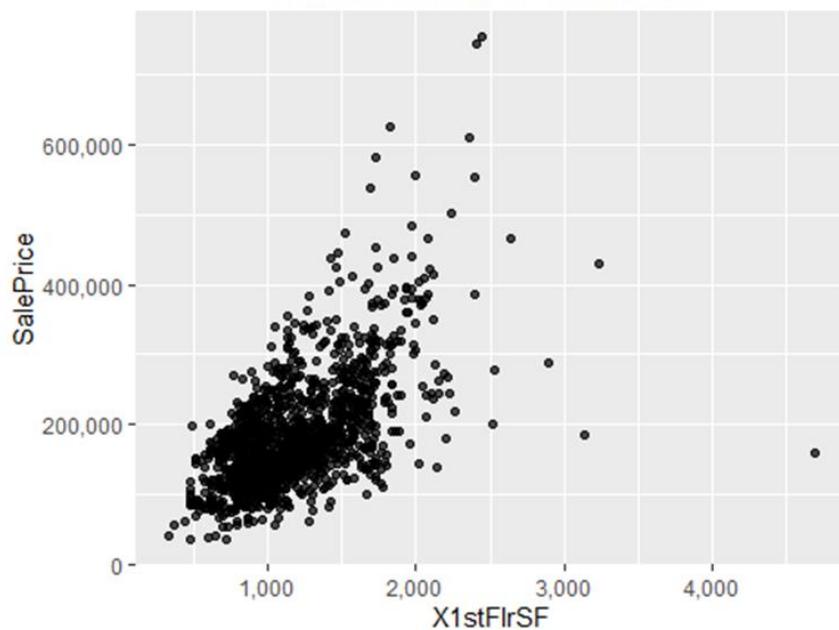
Prilikom analize promenljive TotalBsmtSF, uočen je jedan ekstremni slučaj - podrum površine preko 6000, što značajno odstupa od ostalih podataka. Ovaj red ćemo ukloniti iz skupa podataka.

```
data.train %>% filter(TotalBsmtSF > 6000) %>% select(Order)

##      Order
## 1 1499

ggplot(data.train, aes(x = X1stFlrSF, y = SalePrice)) +
  geom_point(alpha = 0.7) +
  scale_x_continuous(labels = comma) +
  scale_y_continuous(labels = comma) +
  labs(
    title = "Odnos X1stFlrSF i SalePrice",
    x = "X1stFlrSF",
    y = "SalePrice"
  ) +
  theme(plot.title = element_text(hjust = 0.5, face = "bold"))
```

### Odnos X1stFlrSF i SalePrice



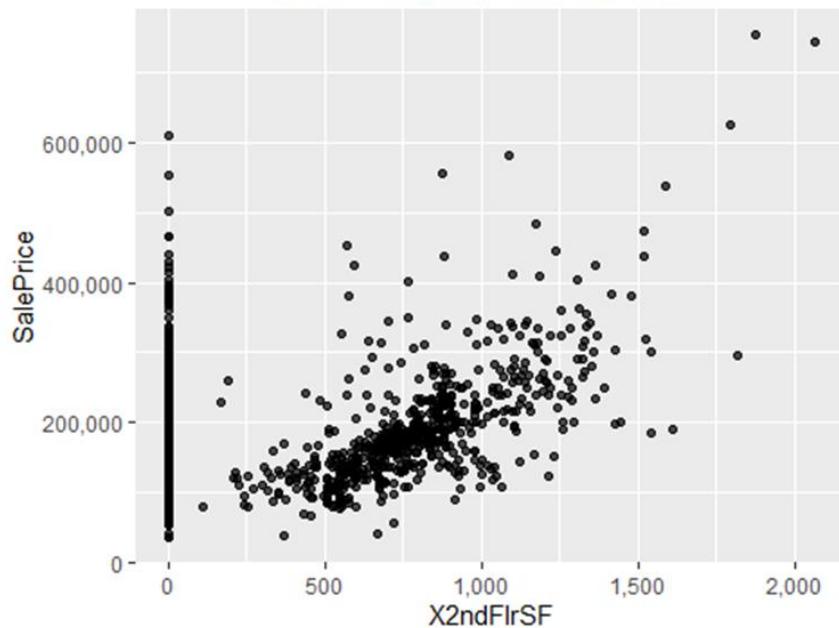
Promenljiva 1stFlrSF pokazuje da jedna kuća ima izuzetno veliku površinu prvog sprata (>4000) i značajno odstupa od ostatka podataka. Tu vrednost ćemo ukloniti.

```
data.train %>% filter(X1stFlrSF > 4500) %>% select(Order)

##      Order
## 1 1499

ggplot(data.train, aes(x = X2ndFlrSF, y = SalePrice)) +
  geom_point(alpha = 0.7) +
  scale_x_continuous(labels = comma) +
  scale_y_continuous(labels = comma) +
  labs(
    title = "Odnos X2ndFlrSF i SalePrice",
    x = "X2ndFlrSF",
    y = "SalePrice"
  ) +
  theme(plot.title = element_text(hjust = 0.5, face = "bold"))
```

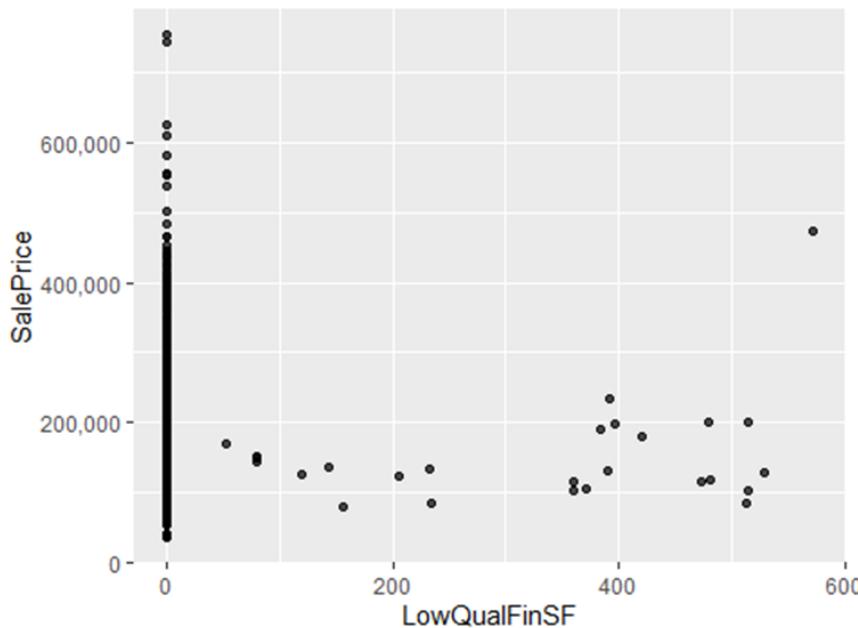
**Odnos X2ndFlrSF i SalePrice**



Na grafiku se može videti odnos cene kuće i površine drugog sprata. Nije uočen nijedan outlier, a primećuje se da određeni broj kuća nema drugi sprat.

```
ggplot(data.train, aes(x = LowQualFinSF, y = SalePrice)) +
  geom_point(alpha = 0.7) +
  scale_x_continuous(labels = comma) +
  scale_y_continuous(labels = comma) +
  labs(
    title = "Odnos LowQualFinSF i SalePrice",
    x = "LowQualFinSF",
    y = "SalePrice"
  ) +
  theme(plot.title = element_text(hjust = 0.5, face = "bold"))
```

### Odnos LowQualFinSF i SalePrice



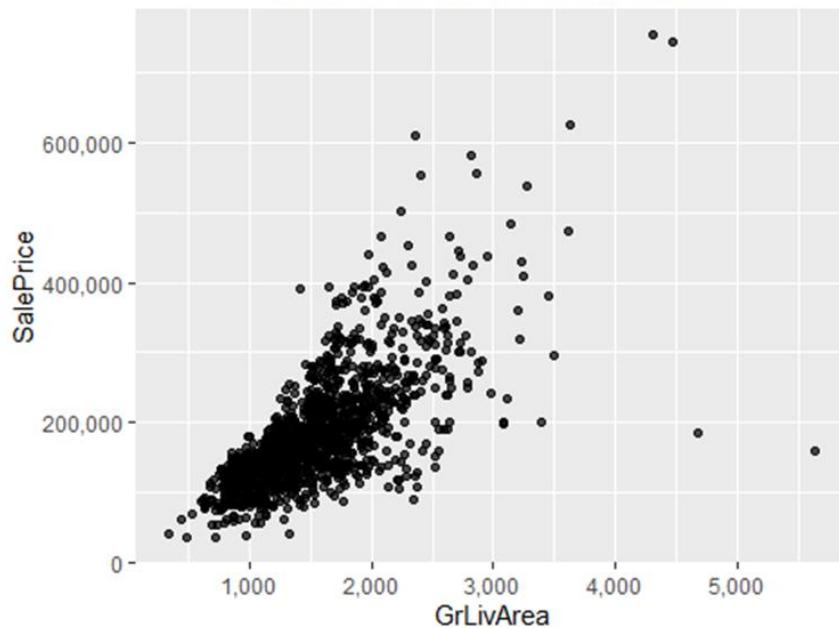
Promenljiva LowQualFinSF pokazuje da jedna kuća ima izuzetno visok broj kvadrata završne obrade niskog kvaliteta i visoku cenu i značajno odstupa od ostalih podataka. Iz tih razloga ćemo je ukloniti.

```
data.train %>% filter(LowQualFinSF > 500 & SalePrice > 400000) %>%
  select(-Order)

##      Order
## 1 2667

ggplot(data.train, aes(x = GrLivArea, y = SalePrice)) +
  geom_point(alpha = 0.7) +
  scale_x_continuous(labels = comma) +
  scale_y_continuous(labels = comma) +
  labs(
    title = "Odnos GrLivArea i SalePrice",
    x = "GrLivArea",
    y = "SalePrice"
  ) +
  theme(plot.title = element_text(hjust = 0.5, face = "bold"))
```

### Odnos GrLivArea i SalePrice



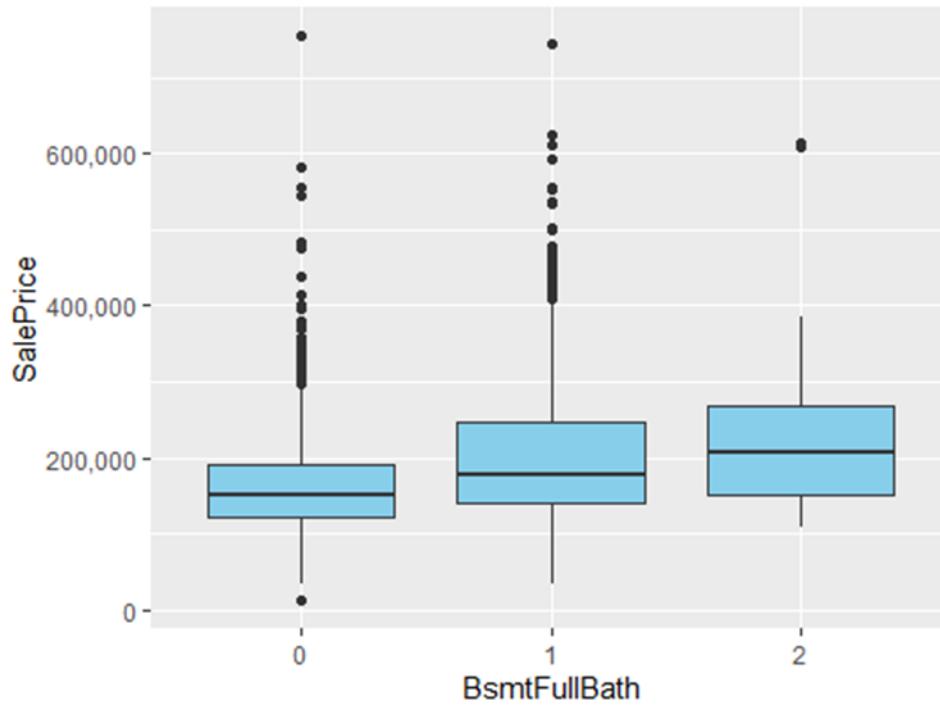
Za ukupnu završenu površinu iznad zemlje (GrLivArea) uočene su dve kuće koje imaju izuzetno velike površine, a nisku cenu i njih ćemo ukloniti.

```
data.train %>% filter(GrLivArea > 4500 & SalePrice < 200000) %>%
  select(Order)

##      Order
## 1    1499
## 2    2181

ggplot(data.train, aes(x = as.factor(BsmtFullBath), y = SalePrice)) +
  geom_boxplot(fill = "skyblue") +
  scale_y_continuous(labels = comma) +
  labs(
    title = "Distribucija SalePrice po BsmtFullBath",
    x = "BsmtFullBath",
    y = "SalePrice"
  ) +
  theme(plot.title = element_text(hjust = 0.5, face = "bold"))
```

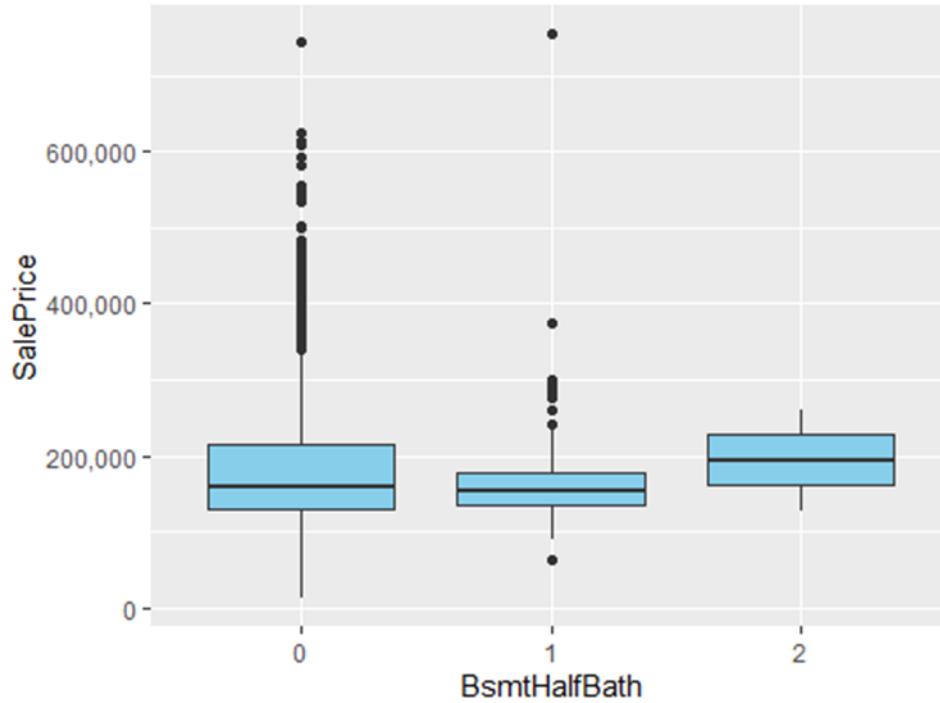
### Distribucija SalePrice po BsmtFullBath



Na grafiku se može videti odnos između ukupnog broja kupatila u podrumu i ciljne promenljive.

```
ggplot(data.train, aes(x = as.factor(BsmtHalfBath), y = SalePrice)) +  
  geom_boxplot(fill = "skyblue") +  
  scale_y_continuous(labels = comma) +  
  labs(  
    title = "Distribucija SalePrice po BsmtHalfBath",  
    x = "BsmtHalfBath",  
    y = "SalePrice"  
  ) +  
  theme(plot.title = element_text(hjust = 0.5, face = "bold"))
```

### Distribucija SalePrice po BsmtHalfBath



Prilikom analize promenljive - podrumska polukupatila (BsmtHalfBath), identifikovane su nekretnine sa ekstremnim vrednostima: tri nekretnine imaju po 2 polukupatila, a jedna nekretnina sa jednim polukupatilom ima vrlo visoku cenu. Uklonićemo vrednosti.

```
data.train %>% filter(BsmtHalfBath == 2) %>% select(Order)

##      Order
## 1 1743
## 2 2821
## 3 2499

data.train %>% filter(BsmtHalfBath == 1 & SalePrice > 700000) %>%
select(Order)

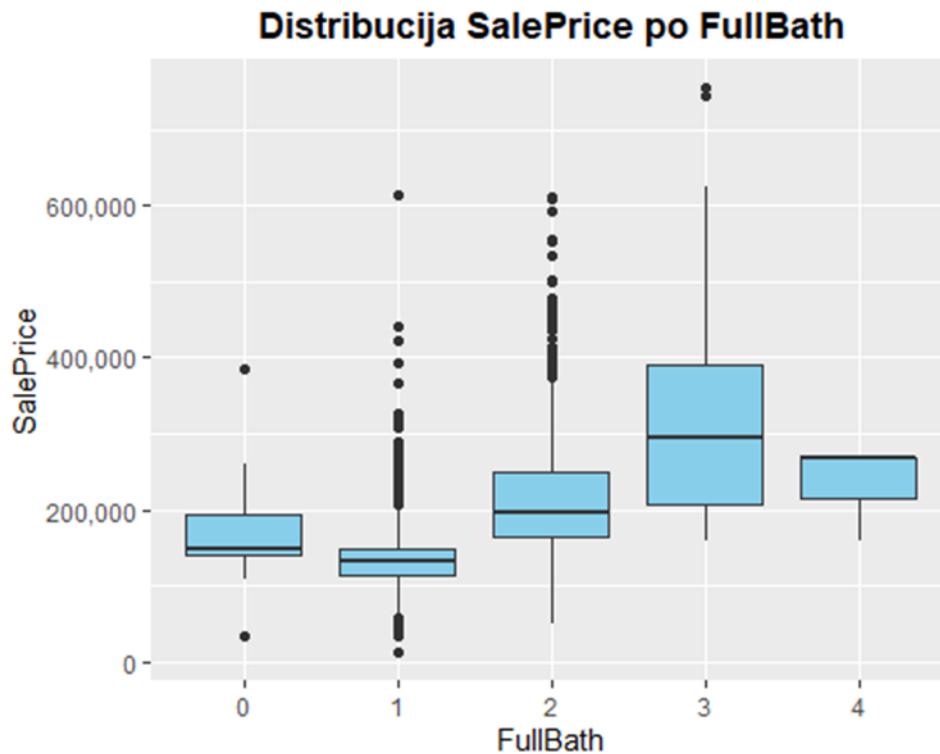
##      Order
## 1 1768

ggplot(data.train, aes(x = as.factor(FullBath), y = SalePrice)) +
  geom_boxplot(fill = "skyblue") +
  scale_y_continuous(labels = comma) +
  labs(
```

```

    title = "Distribucija SalePrice po FullBath",
    x = "FullBath",
    y = "SalePrice"
) +
theme(plot.title = element_text(hjust = 0.5, face = "bold"))

```



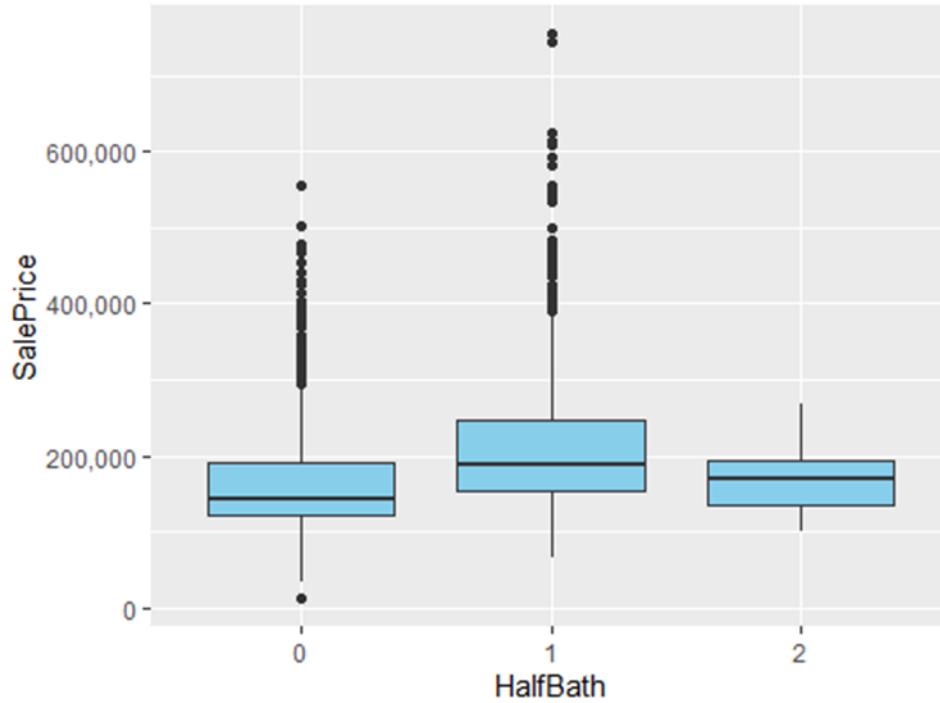
Na grafiku je predstavljen odnos između cene kuće i broja kupatila. Može se zaključiti da nema odstupanja.

```

ggplot(data.train, aes(x = as.factor(HalfBath), y = SalePrice)) +
  geom_boxplot(fill = "skyblue") +
  scale_y_continuous(labels = comma) +
  labs(
    title = "Distribucija SalePrice po HalfBath",
    x = "HalfBath",
    y = "SalePrice"
) +
  theme(plot.title = element_text(hjust = 0.5, face = "bold"))

```

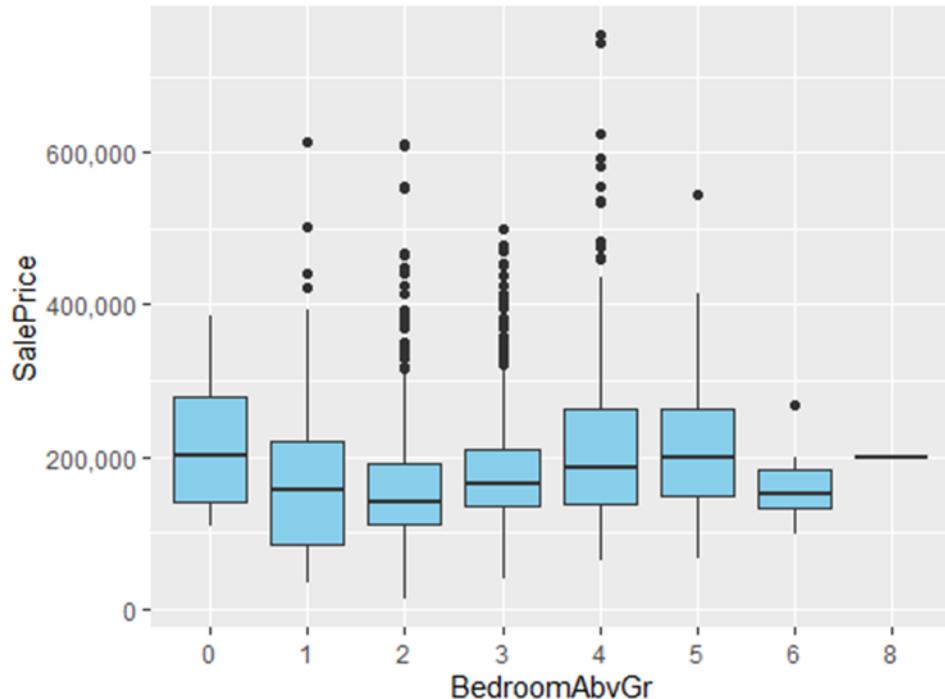
### Distribucija SalePrice po HalfBath



Prilikom analize odnosa cene kuće i broja polukupatila nije uočena nijedna vrednost koja značajno odstupa od ostatka podataka.

```
ggplot(data.train, aes(x = as.factor(BedroomAbvGr), y = SalePrice)) +  
  geom_boxplot(fill = "skyblue") +  
  scale_y_continuous(labels = comma) +  
  labs(  
    title = "Distribucija SalePrice po BedroomAbvGr",  
    x = "BedroomAbvGr",  
    y = "SalePrice"  
  ) +  
  theme(plot.title = element_text(hjust = 0.5, face = "bold"))
```

### Distribucija SalePrice po BedroomAbvGr



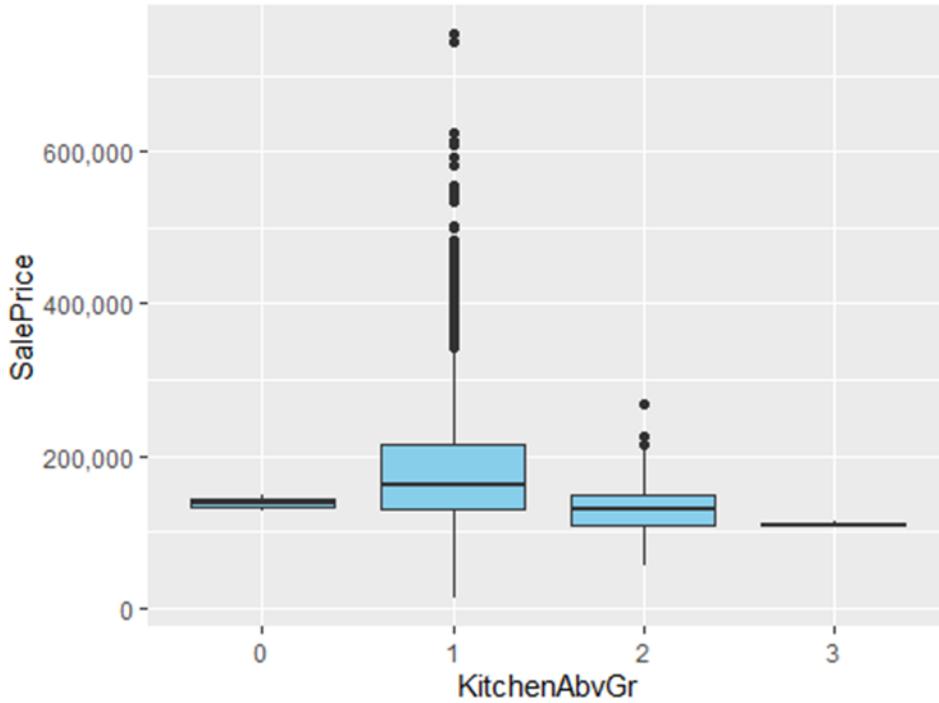
Prilikom analize promenljive koja predstavlja broj spavačih soba iznad zemlje (BedroomAbvGr), uočena je jedna nekretnina sa 8 spavačih soba, što značajno odstupa od većine podataka i nju ćemo ukloniti.

```
Data.train %>% filter(BedroomAbvGr == 8) %>% select(Order)

##      Order
## 1 2195

ggplot(data.train, aes(x = as.factor(KitchenAbvGr), y = SalePrice)) +
  geom_boxplot(fill = "skyblue") +
  scale_y_continuous(labels = comma) +
  labs(
    title = "Distribucija SalePrice po KitchenAbvGr",
    x = "KitchenAbvGr",
    y = "SalePrice"
  ) +
  theme(plot.title = element_text(hjust = 0.5, face = "bold"))
```

## Distribucija SalePrice po KitchenAbvGr



Prilikom analize promenljive koja predstavlja broj kuhinja iznad zemlje (KitchenAbvGr), uočene su dve kuće sa 3 kuhinje i dve kuće bez kuhinje i njih ćemo ukloniti.

```
data.train %>% filter(KitchenAbvGr == 3) %>% select(Order)

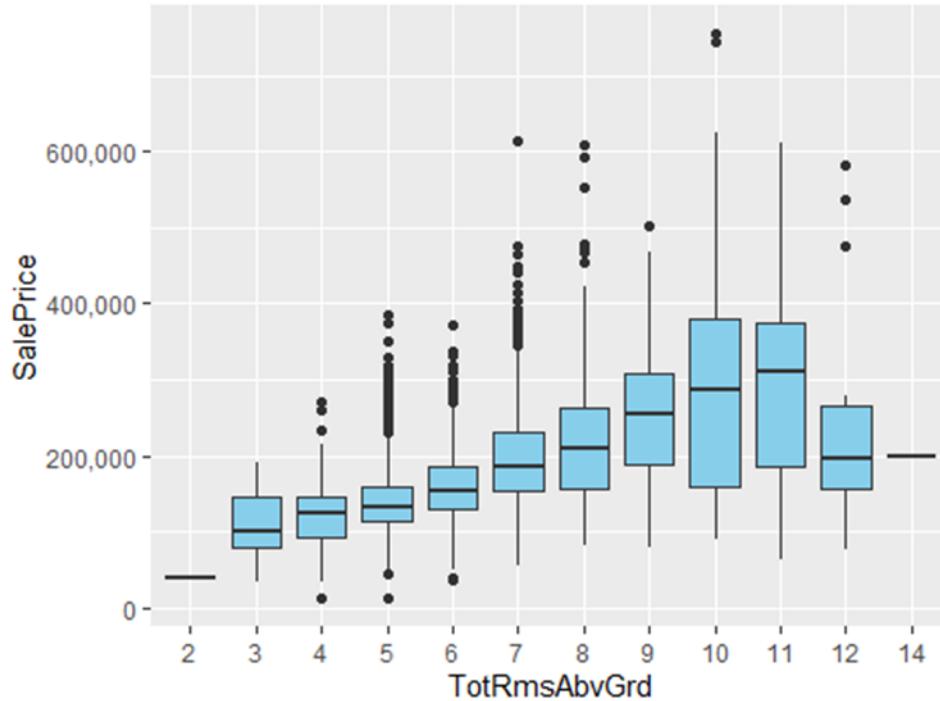
##      Order
## 1    713
## 2    716

data.train %>% filter(KitchenAbvGr == 0) %>% select(Order)

##      Order
## 1  2821
## 2  2254

ggplot(data.train, aes(x = as.factor(TotRmsAbvGrd), y = SalePrice)) +
  geom_boxplot(fill = "skyblue") +
  scale_y_continuous(labels = comma) +
  labs(
    title = "Distribucija SalePrice po TotRmsAbvGrd",
    x = "TotRmsAbvGrd",
    y = "SalePrice"
  ) +
  theme(plot.title = element_text(hjust = 0.5, face = "bold"))
```

## Distribucija SalePrice po TotRmsAbvGrd



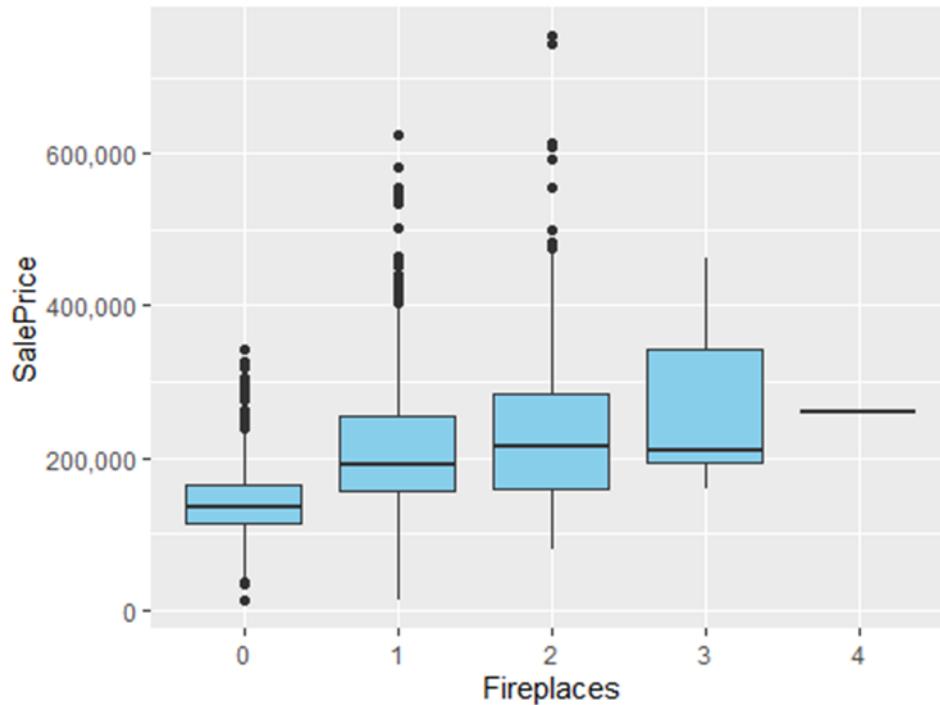
Prilikom pregleda broja soba iznad zemlje (TotRmsAbvGrd), uočena je jedna kuća sa 14 soba, što odstupa od ostalih podataka. Ova vrednost će biti uklonjena.

```
data.train %>% filter(TotRmsAbvGrd == 14) %>% select(Order)

##      Order
## 1 2195

ggplot(data.train, aes(x = as.factor(Fireplaces), y = SalePrice)) +
  geom_boxplot(fill = "skyblue") +
  scale_y_continuous(labels = comma) +
  labs(
    title = "Distribucija SalePrice po Fireplaces",
    x = "Fireplaces",
    y = "SalePrice"
  ) +
  theme(plot.title = element_text(hjust = 0.5, face = "bold"))
```

### Distribucija SalePrice po Fireplaces

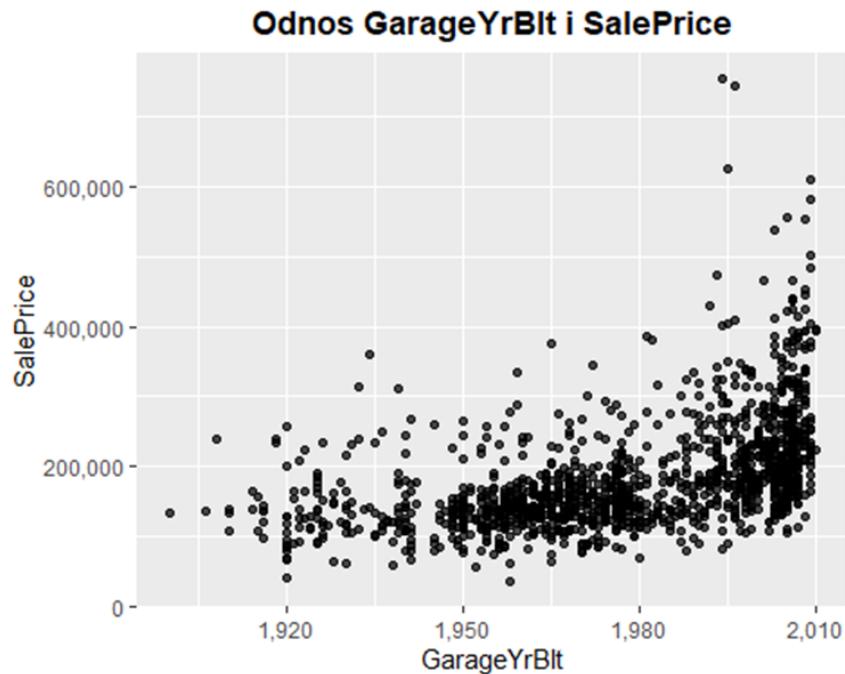


Grafik predstavlja odnos između cene kuće i broja kamina. Uočena je jedna vrednost gde je broj kamina jednak 4 i tu vrednost ćemo ukloniti.

```
data.train %>% filter(Fireplaces == 4) %>% select(Order)

##   Order
## 1 2499

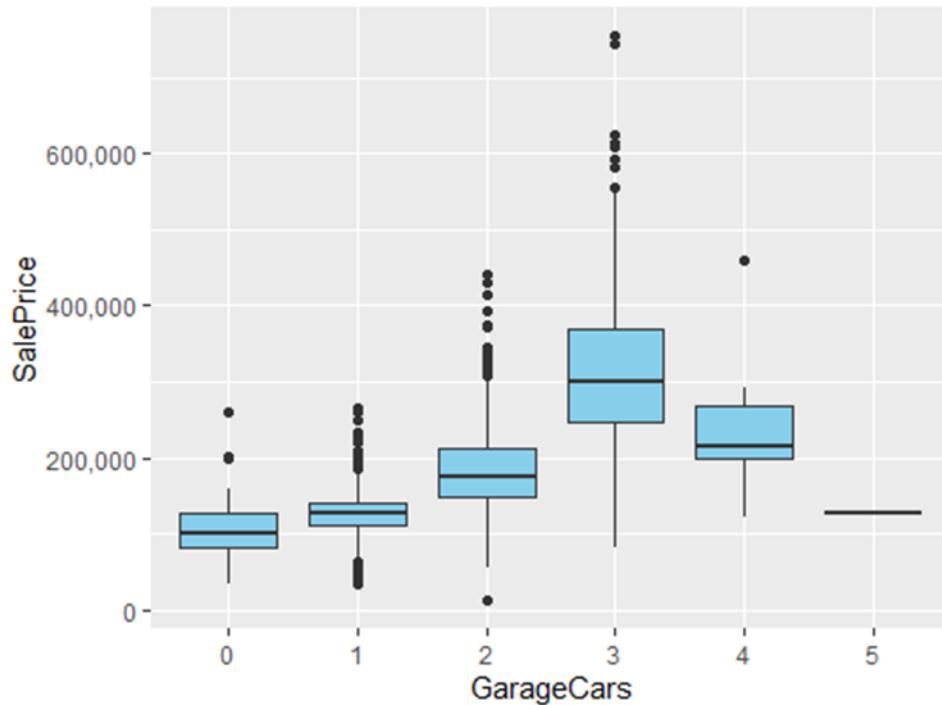
ggplot(data.train, aes(x = GarageYrBlt, y = SalePrice)) +
  geom_point(alpha = 0.7) +
  scale_x_continuous(labels = comma) +
  scale_y_continuous(labels = comma) +
  labs(
    title = "Odnos GarageYrBlt i SalePrice",
    x = "GarageYrBlt",
    y = "SalePrice"
  ) +
  theme(plot.title = element_text(hjust = 0.5, face = "bold"))
```



Grafik zavisnosti cene kuće i godine izgradnje garaže prikazuje da nema vrednosti koje značajnije odstupaju od ostatka podataka.

```
ggplot(data.train, aes(x = as.factor(GarageCars), y = SalePrice)) +
  geom_boxplot(fill = "skyblue") +
  scale_y_continuous(labels = comma) +
  labs(
    title = "Distribucija SalePrice po GarageCars",
    x = "GarageCars",
    y = "SalePrice"
  ) +
  theme(plot.title = element_text(hjust = 0.5, face = "bold"))
```

### Distribucija SalePrice po GarageCars

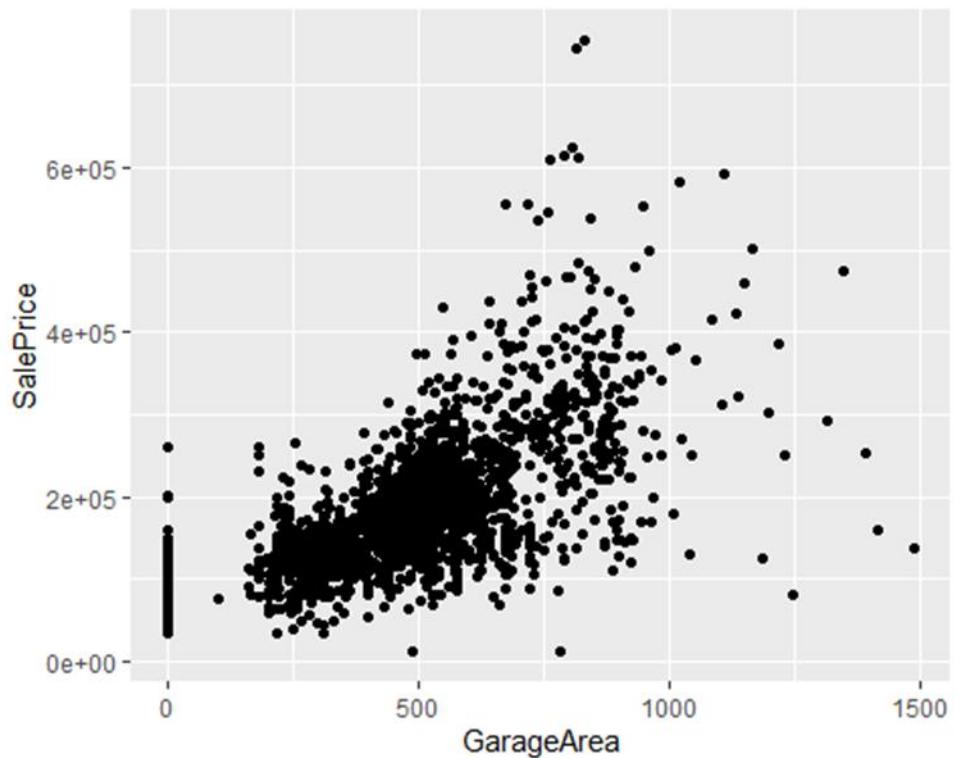


Grafik prikazuje distribuciju cene kuće po kapacitetu garaže. Na grafiku je primećena vrednost gde je kapacitet garaže 5. Tu vrednost ćemo ukloniti.

```
data.train %>% filter(GarageCars == 5) %>% select(Order)

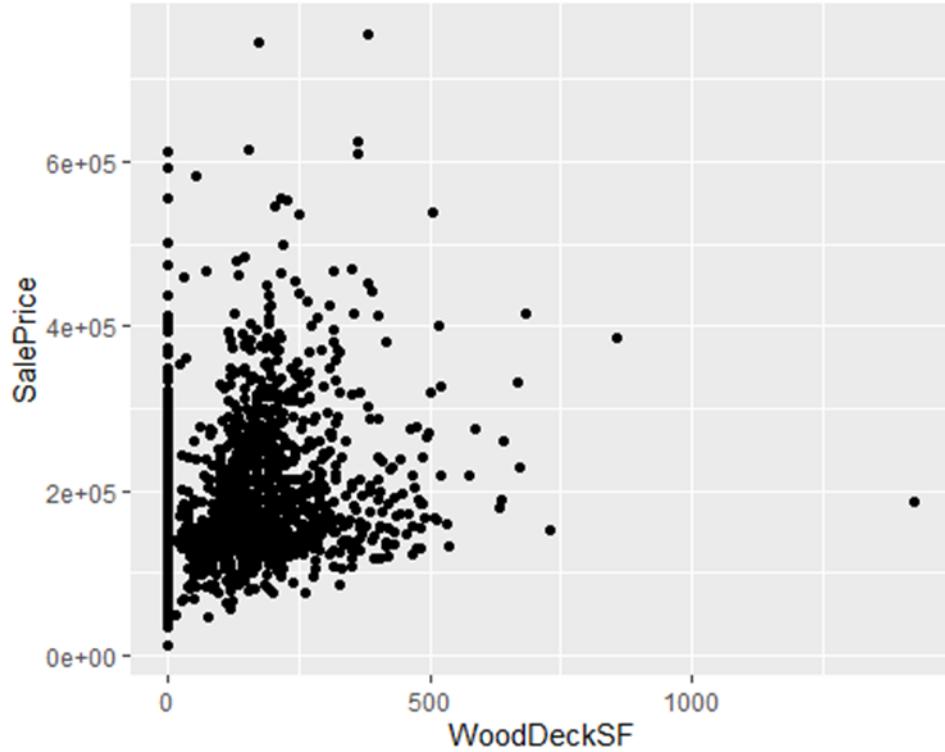
##   Order
## 1    747

ggplot(data.train, aes(x = GarageArea, y = SalePrice)) +
  geom_point(alpha = 0.7) +
  scale_x_continuous(labels = comma) +
  scale_y_continuous(labels = comma) +
  labs(
    title = "Odnos GarageArea i SalePrice",
    x = "GarageArea",
    y = "SalePrice"
  ) +
  theme(plot.title = element_text(hjust = 0.5, face = "bold"))
```



Grafik predstavlja zavisnost površine garaže i ciljne promenljive.

```
ggplot(data.train, aes(x = WoodDeckSF, y = SalePrice)) +
  geom_point(alpha = 0.7) +
  scale_x_continuous(labels = comma) +
  scale_y_continuous(labels = comma) +
  labs(
    title = "Odnos WoodDeckSF i SalePrice",
    x = "WoodDeckSF",
    y = "SalePrice"
  ) +
  theme(plot.title = element_text(hjust = 0.5, face = "bold"))
```



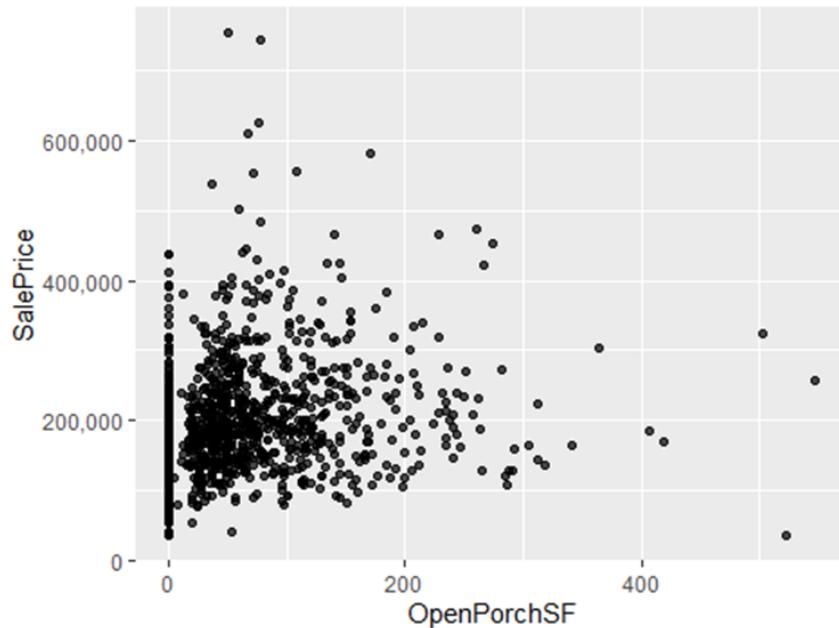
Na grafiku zavisnosti cene kuće i površine drvene terase nisu uočeni outlieri.

```
data.train %>% filter(WoodDeckSF > 1000) %>% select(Order)

##   Order
## 1 2294

ggplot(data.train, aes(x = OpenPorchSF, y = SalePrice)) +
  geom_point(alpha = 0.7) +
  scale_x_continuous(labels = comma) +
  scale_y_continuous(labels = comma) +
  labs(
    title = "Odnos OpenPorchSF i SalePrice",
    x = "OpenPorchSF",
    y = "SalePrice"
  ) +
  theme(plot.title = element_text(hjust = 0.5, face = "bold"))
```

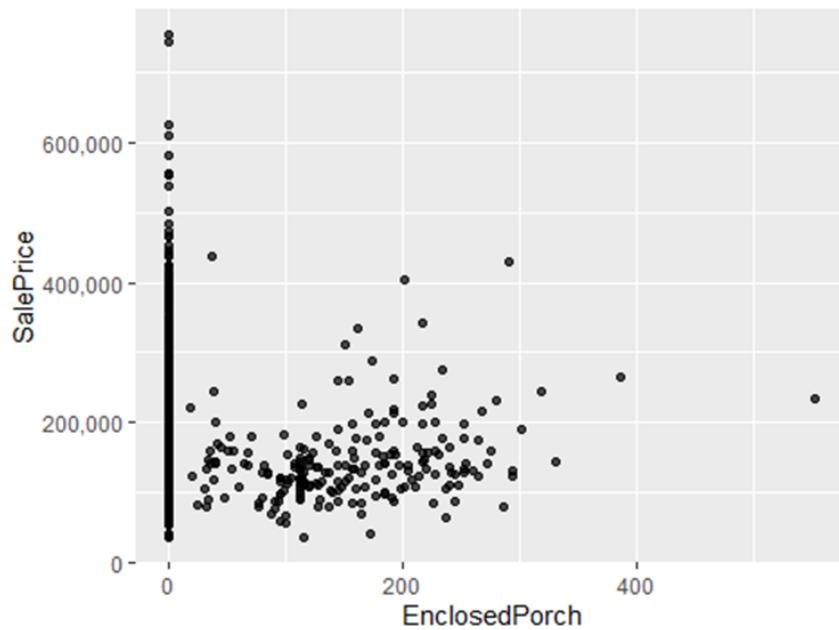
### Odnos OpenPorchSF i SalePrice



Prilikom analize površine otvorene terase (OpenPorchSF), uočena su dva ekstremna slučaja za EnclosedPorch > 520. Vrednost će biti uklonjena.

```
data.train %>% filter(OpenPorchSF > 520 & SalePrice < 100000) %>%  
select(Order)  
  
##      Order  
## 1 2066  
## 2 727  
  
ggplot(data.train, aes(x = EnclosedPorch, y = SalePrice)) +  
  geom_point(alpha = 0.7) +  
  scale_x_continuous(labels = comma) +  
  scale_y_continuous(labels = comma) +  
  labs(  
    title = "Odnos EnclosedPorch i SalePrice",  
    x = "EnclosedPorch",  
    y = "SalePrice"  
) +  
  theme(plot.title = element_text(hjust = 0.5, face = "bold"))
```

### Odnos EnclosedPorch i SalePrice



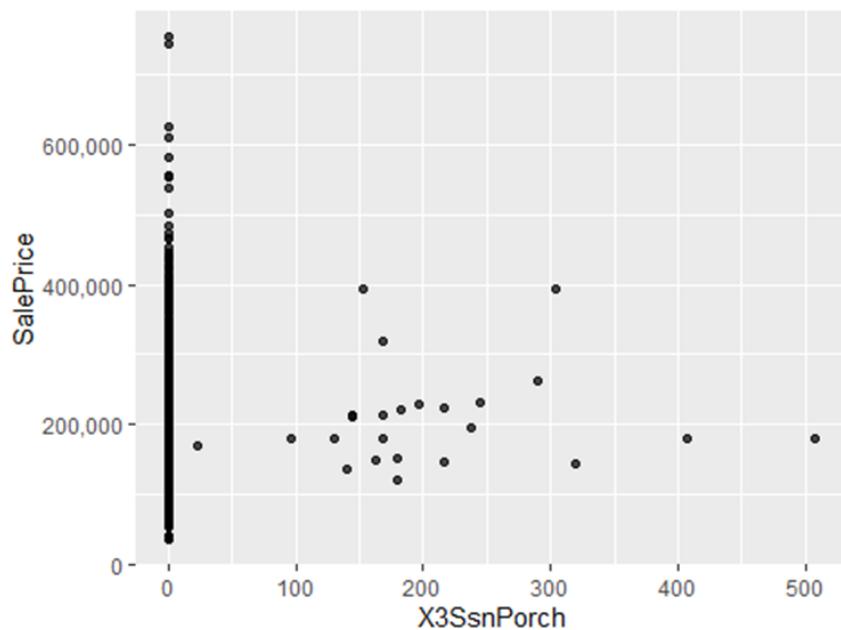
Prilikom analize površine zatvorene terase (EnclosedPorch), uočen je jedan ekstremni slučaj ( $\text{EnclosedPorch} > 1000$ ) koji ima visoku polugu. Ovu vrednost ćemo ukloniti.

```
data.train %>% filter(EnclosedPorch > 1000) %>% select(Order)

##      Order
## 1 2090

ggplot(data.train, aes(x = X3SsnPorch, y = SalePrice)) +
  geom_point(alpha = 0.7) +
  scale_x_continuous(labels = comma) +
  scale_y_continuous(labels = comma) +
  labs(
    title = "Odnos X3SsnPorch i SalePrice",
    x = "X3SsnPorch",
    y = "SalePrice"
  ) +
  theme(plot.title = element_text(hjust = 0.5, face = "bold"))
```

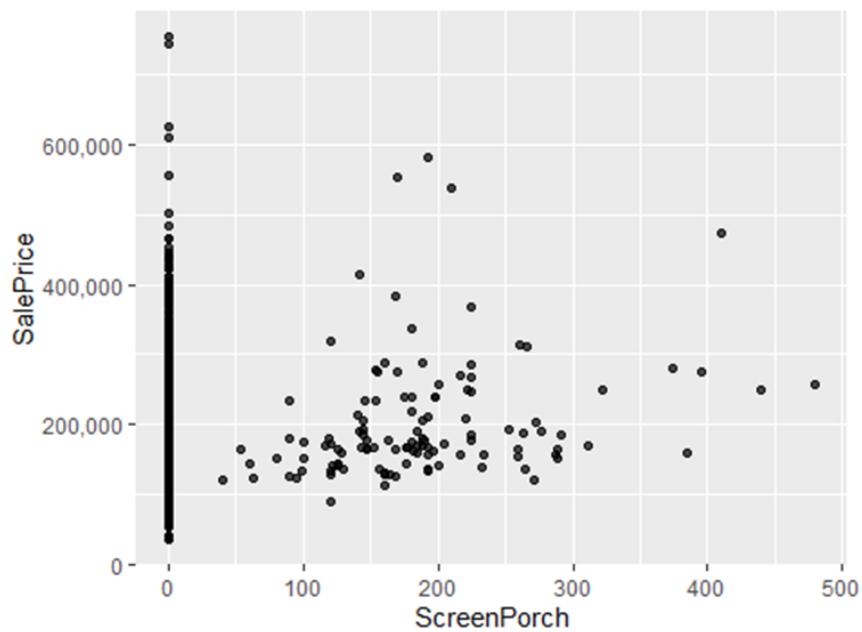
**Odnos X3SsnPorch i SalePrice**



Na grafiku je predstavljen odnos cene kuće i površine trosezonske terase. Primećuje se da većina kuća nema trosezonsku terasu. Ostali deo podataka ne sadrži outliere.

```
ggplot(data.train, aes(x = ScreenPorch, y = SalePrice)) +  
  geom_point(alpha = 0.7) +  
  scale_x_continuous(labels = comma) +  
  scale_y_continuous(labels = comma) +  
  labs(  
    title = "Odnos ScreenPorch i SalePrice",  
    x = "ScreenPorch",  
    y = "SalePrice"  
  ) +  
  theme(plot.title = element_text(hjust = 0.5, face = "bold"))
```

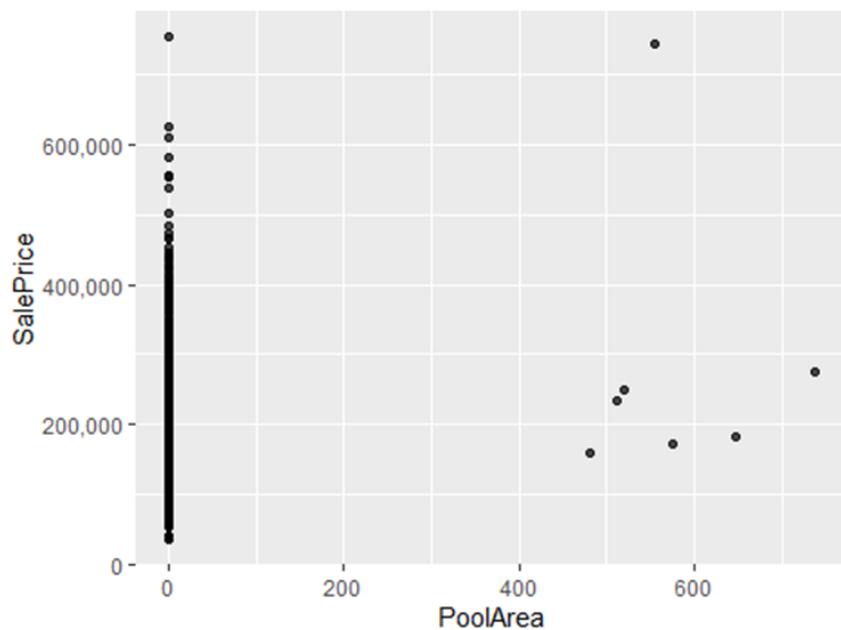
Odnos ScreenPorch i SalePrice



Na grafiku je predstavljen odnos cene kuće i površine mrežaste terase. Primećuje se da dobar deo kuća nema mrežastu terasu. Ostali deo podataka ne sadrži outliere.

```
ggplot(data.train, aes(x = PoolArea, y = SalePrice)) +
  geom_point(alpha = 0.7) +
  scale_x_continuous(labels = comma) +
  scale_y_continuous(labels = comma) +
  labs(
    title = "Odnos PoolArea i SalePrice",
    x = "PoolArea",
    y = "SalePrice"
  ) +
  theme(plot.title = element_text(hjust = 0.5, face = "bold"))
```

Odnos PoolArea i SalePrice



Na grafiku je predstavljen odnos cene kuće i površine bazena. Može se primetiti da samo nekoliko kuća poseduje bazen.

Sada smo sakupili Order vrednost svih outliera i uklonićemo ih iz dataseta.

```
values = c(1499, 1266, 2072, 1571, 957, 2116, 2181, 1734, 2499, 1768, 18,
         1761, 2667, 2821, 2195, 713, 716, 2254, 926, 747, 2294, 2066, 727,
         2090)

data.train <- data.train %>%
  filter(!(Order %in% values))
```

## Uklanjanje NA vrednosti

Za početak ćemo spojiti train i test skup kako bismo uklonili sve NA vrednosti.

```
data.test$SalePrice <- NA  
data <- rbind(data.train, data.test)
```

Kolone sa NA vrednostima:

```
na_columns = sort(colSums(is.na(data))[colSums(is.na(data)) > 0])  
na_columns  
  
## BsmtFinSF1 BsmtFinSF2 BsmtUnfSF TotalBsmtSF GarageCars GarageA  
rea  
## 1 1 1 1 1 1  
## BsmtFullBath BsmtHalfBath MasVnrArea BsmtQual BsmtCond BsmtExpos  
ure  
## 2 2 23 76 76 76  
## BsmtFinType1 BsmtFinType2 GarageType GarageFinish GarageQual GarageC  
ond  
## 76 76 155 155 156 156  
## GarageYrBlt LotFrontage FireplaceQu Fence Alley MiscFeat  
ure  
## 157 488 1403 2335 2707 2  
## PoolQC  
## 2892
```

## Electrical - električni sistem

- Ima jednu NA
- Kategoriska, nominalna

```
tbl <- xtabs(~Electrical, data = data)
most_common <- names(which.max(tbl))
data$Electrical[is.na(data$Electrical)] <- most_common
```

**MasVnrType i MasVnrArea - tip i površina zida koji je pokriven dekorativnim materijalom**

- MasVnrType
  - Kategorijска, nominalna
  - 23 NA

Popunjavamo najčešćom vrednošću.

```
tbl <- xtabs(~MasVnrType, data = data)
most_common <- names(which.max(tbl))
data$MasVnrType[is.na(data$MasVnrType)] <- most_common
```

- MasVnrArea
  - Numeričkog tipa
  - 23 NA

Uradićemo Shapiro test da bismo utvrdili kakva je raspodela.

```
shapiro.test(data$MasVnrArea)

##
##  Shapiro-Wilk normality test
##
## data: data$MasVnrArea
## W = 0.63722, p-value < 2.2e-16
```

Pošto test ukazuje da raspodela nije normalna, koristimo medijanu da popunimo NA vrednosti.

```
data_median = median(data$MasVnrArea[!is.na(data$MasVnrArea)])
data$MasVnrArea[is.na(data$MasVnrArea)] <- data_median
```

## Bsmt - kolone koje opisuju podrum

NA znači da ne postoji podrum.

- BsmtCond - stanje podruma
  - Kategorijiska, ordinalna
  - 76 NA
- BsmtExposure - da li podrum ima prozore ili vrata koja izlaze u dvorište
  - Kategorijiska, ordinalna
  - 76 NA
- Bsmt FinType2 - kvalitet druge završene površine podruma (ako postoji)
  - Kategorijiska, ordinalna
  - 76 NA
- BsmtQual - visina podruma
  - Kategorijiska, ordinalna
  - 76 NA
- Bsmt FinType1 - kvalitet zavrsene povrsine podruma
  - Kategorijiska, ordinalna
  - 76 NA

Postoji jedan red gde su BsmtFinSF1, BsmtFinSF2, BsmtUnfSF i TotalBsmtSF NA, to ćemo popuniti nulama.

```
data$BsmtFinSF1[is.na(data$BsmtFinSF1)] <- 0
data$BsmtFinSF2[is.na(data$BsmtFinSF2)] <- 0
data$BsmtUnfSF[is.na(data$BsmtUnfSF)] <- 0
data$TotalBsmtSF[is.na(data$TotalBsmtSF)] <- 0

data$BsmtQual[data$BsmtQual == ""] <- "NoBasement"
data$BsmtCond[data$BsmtCond == ""] <- "NoBasement"
data$BsmtExposure[data$BsmtExposure == ""] <- "NoBasement"
data$BsmtFinType1[data$BsmtFinType1 == ""] <- "NoBasement"
data$BsmtFinType2[data$BsmtFinType2 == ""] <- "NoBasement"
```

- BsmtFullBath, BsmtHalfBath - Broj kupatila/polu-kupatila u podrumu
  - Numeričkog tipa
  - 2 NA
  - U ovim redovima ne postoji podrum, pa ćemo popuniti nulama.

```

data$BsmtFullBath[is.na(data$BsmtFullBath)] <- 0
data$BsmtHalfBath[is.na(data$BsmtHalfBath)] <- 0

data$BsmtQual[is.na(data$BsmtQual)] <- "NoBasement"
data$BsmtCond[is.na(data$BsmtCond)] <- "NoBasement"
data$BsmtFinType1[is.na(data$BsmtFinType1)] <- "NoBasement"
data$BsmtFinType2[is.na(data$BsmtFinType2)] <- "NoBasement"
data$BsmtExposure[is.na(data$BsmtExposure)] <- "NoBasement"

```

### **Garage - kolone koje opisuju garažu**

NA znači da ne postoji garaža.

- GarageType - tip garaže
  - Kategoriskska, nominalna
  - 155 NA
- GarageYrBlt - godina kada je garaža izgrađena
  - Numerička
  - 157 NA
- GarageFinish - unutrašnja završna obrada garaže
  - Kategoriskska, ordinalna
  - 155 NA
- GarageQual - kvalitet garaže
  - Kategoriskska, ordinalna
  - 156 NA
- GarageCond - stanje garaže
  - Kategoriskska, ordinalna
  - 156 NA

Postoji jedan red u test skupu gde je godina izgranje garaže 2207. Očigledno je da je došlo do greške.

Takođe, ta kuća je izgrađena 2006, a renovirana 2007. godine, pa ćemo vrednost GarageYrBlt zameniti godinom renoviranja.

```

data$GarageYrBlt[data$GarageYrBlt==2207] <- 2007

```

Postoji jedan red gde je GarageType = “Detachd”, a svi ostali podaci o garaži su NA.

```

data$GarageArea[data$Order == 2237] <- 0
data$GarageCars[data$Order == 2237] <- 0
data$GarageType[data$Order == 2237] <- NA
data$GarageQual[data$Order == 2237] <- NA
data$GarageCond[data$Order == 2237] <- NA
data$GarageFinish[data$Order == 2237] <- NA
data$GarageYrBlt[data$Order == 2237] <- -1

```

Postoji red gde su prisutni podaci za GarageType, GarageCars i GarageArea, a svi ostali podaci o garaži su NA.

```

tbl <- xtabs(~ GarageFinish, data = data[!is.na(data$GarageFinish), ])
most_common_finish <- names(which.max(tbl))
data$GarageFinish[data$Order == 1357] <- most_common_finish

tbl <- xtabs(~ GarageQual, data = data[!is.na(data$GarageQual), ])
most_common_qual <- names(which.max(tbl))
data$GarageQual[data$Order == 1357] <- most_common_qual

tbl <- xtabs(~ GarageCond, data = data[!is.na(data$GarageCond), ])
most_common_cond <- names(which.max(tbl))
data$GarageCond[data$Order == 1357] <- most_common_cond

data$GarageYrBlt[data$Order == 1357] <- data$YearBuilt[data$Order == 1357]

data$GarageType[is.na(data$GarageType)] <- "NoGarage"
data$GarageFinish[is.na(data$GarageFinish)] <- "NoGarage"
data$GarageQual[is.na(data$GarageQual)] <- "NoGarage"
data$GarageCond[is.na(data$GarageCond)] <- "NoGarage"
data$GarageYrBlt[is.na(data$GarageYrBlt)] <- -1

```

## LotFrontage - dužina placa koja se graniči sa ulicom

- Numeričkog tipa
- Koristimo medijanu za svaki Neighborhood kako bismo popunili NA vrednosti.
- 488 NA

Tri vrednosti ostaju nepotpunjene, jer za su za taj Neighborhood sve vrednosti NA. Popunićemo ih medijanom celog skupa.

```

data <- data %>%
  group_by(Neighborhood) %>%
  mutate(LotFrontage = ifelse(is.na(LotFrontage),
                             median(LotFrontage, na.rm = TRUE),
                             LotFrontage)) %>%
  ungroup()

data$LotFrontage[is.na(data$LotFrontage)] <- median(data$LotFrontage, na.rm = TRUE)

data$LotFrontage <- as.integer(data$LotFrontage)

```

### FireplaceQu - kvalitet kamina

- Kategorijkska, ordinalna
- NA znači da nema kamina.
- 1403 NA

```

data$FireplaceQu[is.na(data$FireplaceQu)] <- "NoFireplace"

```

### Fence - kvalitet ograde

- Kategorijkska, nominalna
- NA znači da nema ograde.
- 2335 NA

```

data$Fence[is.na(data$Fence)] <- "NoFence"

```

### Alley - zadnji prilaz sa puta

- Kategorijkska, nominalna
- NA znači da nema prilaz
- 2707 NA

```

data$Alley[is.na(data$Alley)] <- "NoAccess"

```

## MiscFeature - dodatni featuri

- Kategorijkska, ordinalna
- NA znači da nema dodatnih featura.

Postoji kolona MiscVal koja predstavlja vrednost MiscFeature-a koji postoje, tako da ćemo ukloniti kolonu MiscFeature, a ostaviti MiscVal.

```
data$MiscFeature <- NULL
```

## PoolQC - kvalitet bazena

- Kategorijkska, ordinalna
- NA znači da ne postoji bazen.
- 2892 NA

```
data$PoolQC[is.na(data$PoolQC)] <- "NoPool"
```

Sve kategoriskske promenljive ćemo pretvoriti u faktor promenljive.

```
data$ExterQual <- factor(data$ExterQual, levels = c("Po", "Fa", "TA", "Gd", "Ex"), ordered = TRUE)
data$ExterCond <- factor(data$ExterCond, levels = c("Po", "Fa", "TA", "Gd", "Ex"), ordered = TRUE)
data$BsmtQual <- factor(data$BsmtQual, levels = c("NoBasement", "Po", "Fa", "TA", "Gd", "Ex"), ordered = TRUE)
data$BsmtCond <- factor(data$BsmtCond, levels = c("NoBasement", "Po", "Fa", "TA", "Gd", "Ex"), ordered = TRUE)
data$HeatingQC <- factor(data$HeatingQC, levels = c("Po", "Fa", "TA", "Gd", "Ex"), ordered = TRUE)
data$KitchenQual <- factor(data$KitchenQual, levels = c("Po", "Fa", "TA", "Gd", "Ex"), ordered = TRUE)
data$FireplaceQu <- factor(data$FireplaceQu, levels = c("NoFireplace", "Po", "Fa", "TA", "Gd", "Ex"), ordered = TRUE)
data$GarageQual <- factor(data$GarageQual, levels = c("NoGarage", "Po", "Fa", "TA", "Gd", "Ex"), ordered = TRUE)
data$GarageCond <- factor(data$GarageCond, levels = c("NoGarage", "Po", "Fa", "TA", "Gd", "Ex"), ordered = TRUE)

data$PoolQC <- factor(data$PoolQC, levels = c("NoPool", "Fa", "TA", "Gd", "Ex"), ordered = TRUE)
```

```

data$BsmtExposure <- factor(data$BsmtExposure, levels = c("NoBasement", "No",
  "Mn", "Av", "Gd"), ordered = TRUE)

data$BsmtFinType1 <- factor(data$BsmtFinType1, levels = c("NoBasement", "Unf"
, "LwQ", "Rec", "BLQ", "ALQ", "GLQ"), ordered = TRUE)
data$BsmtFinType2 <- factor(data$BsmtFinType2, levels = c("NoBasement", "Unf"
, "LwQ", "Rec", "BLQ", "ALQ", "GLQ"), ordered = TRUE)

data$Functional <- factor(data$Functional, levels = c("Sal", "Sev", "Maj2", "Maj1",
  "Mod", "Min2", "Min1", "Typ"), ordered = TRUE)

data$GarageFinish <- factor(data$GarageFinish, levels = c("NoGarage", "Unf",
  "RFn", "Fin"), ordered = TRUE)

nominalne <- data %>%
  select(where(is.character)) %>%
  names()
data[nominalne] <- lapply(data[nominalne], factor)

```

Pretvorimo ordinalne kategorijske promenljive u integer kako bi model razlikovao nivoje.

```

data$ExterQual <- as.integer(data$ExterQual)
data$ExterCond <- as.integer(data$ExterCond)
data$BsmtQual <- as.integer(data$BsmtQual)
data$BsmtCond <- as.integer(data$BsmtCond)
data$HeatingQC <- as.integer(data$HeatingQC)
data$KitchenQual <- as.integer(data$KitchenQual)
data$FireplaceQu <- as.integer(data$FireplaceQu)
data$GarageQual <- as.integer(data$GarageQual)
data$GarageCond <- as.integer(data$GarageCond)
data$PoolQC <- as.integer(data$PoolQC)
data$BsmtExposure <- as.integer(data$BsmtExposure)
data$BsmtFinType1 <- as.integer(data$BsmtFinType1)
data$BsmtFinType2 <- as.integer(data$BsmtFinType2)
data$Functional <- as.integer(data$Functional)
data$GarageFinish <- as.integer(data$GarageFinish)

```

Može se primetiti da je MSSubClass numerička, a u stvari predstavlja nominalnu, kategorijsku promenljivu.

```



```

Sada ćemo opet podeliti na test i train skup.

```

data.train <- data[1:nrow(data.train), ]
data.test  <- data[(nrow(data.train)+1):nrow(data), ]

data.test$SalePrice <- NULL

```

## Neobične vrednosti / greške

Postoji jedan red u test skupu gde je godina izgradnje garaže 2207. Očigledno je da je došlo do greške. Takođe, ta kuća je izgrađena 2006, a renovirana 2007. godine, pa ćemo zameniti tu vrednost.

Postoje dva reda gde je YearRemodAdd nakon YrSold. Zamenićemo YearRemodAdd sa YearBuilt.

Mogu se uočiti i dva reda gde je HouseAge manji od 0, što znači da je kuća prodata pre nego što je izgrađena, odnosno da je došlo do greške. Obrisacemo ta dva reda.

Takođe, postoji red gde je YearBuilt > YearRemodAdd. I taj red ćemo obrisati.

```

data$GarageYrBlt[data$GarageYrBlt==2207] <- 2007

data$YearRemodAdd[data$YearRemodAdd > data$YrSold] <- data$YearBuilt[data$YearRemodAdd > data$YrSold]

data <- data %>%
  filter(YrSold - YearBuilt >= 0)
data <- data %>%
  filter(YearBuilt <= YearRemodAdd)

```

## Podela podataka na trening i testni skup

Celokupni skup podataka ćemo podeliti na trening skup i testni skup u odnosu 80:20.

Trening skup nam služi za obuku modela i na njemu se model uči. Bitno je da ovih podataka bude dovoljno kako bi model mogao adekvatno da uoči obrasce koji se javljaju u podacima.

Testni skup nam služi za merenje performansi već formiranog modela. Model ne sme da vidi podatke iz testnog skupa tokom faze treniranja zato što će onda da ima pristrasnost prema tim podacima i njegove performanse biće optimistične.

Cilj je da se model trenira na jednim a testira na popunom drugim podacima koje do sada nije video, zato što je u interesu da model dobro predviđa za podatke koje prvi put vidi, a ne za one koje je već prethodno video.

```

set.seed(123)
train_idx = sample(seq_len(nrow(data)), size = 0.8 * nrow(data))

data.train = data[train_idx, ]
data.test = data[-train_idx, ]

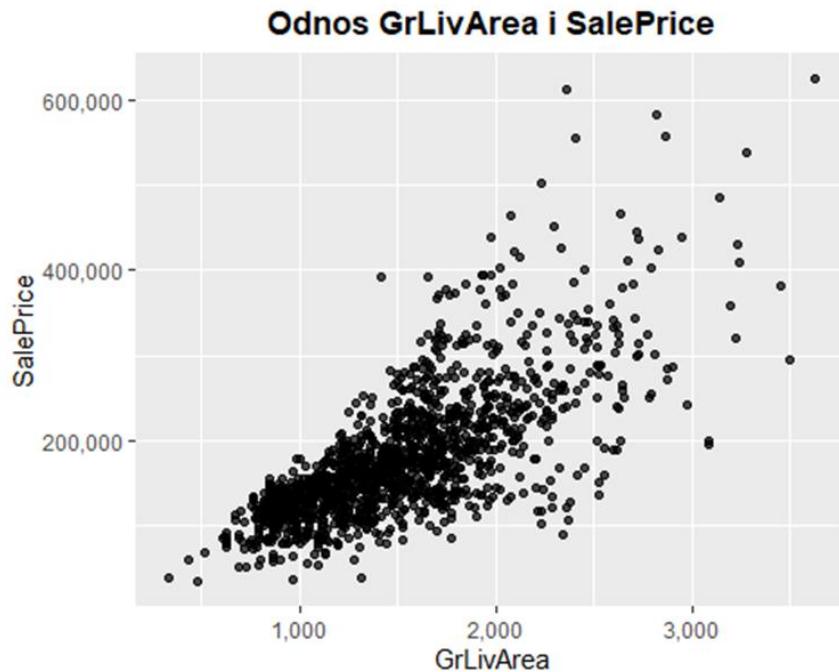
```

## Eksplorativna analiza podataka (EDA)

Prilikom istraživačke analize podataka fokusirali smo se na atribute za koje domensko znanje sugerše da imaju značajan uticaj na cenu nekretnine:

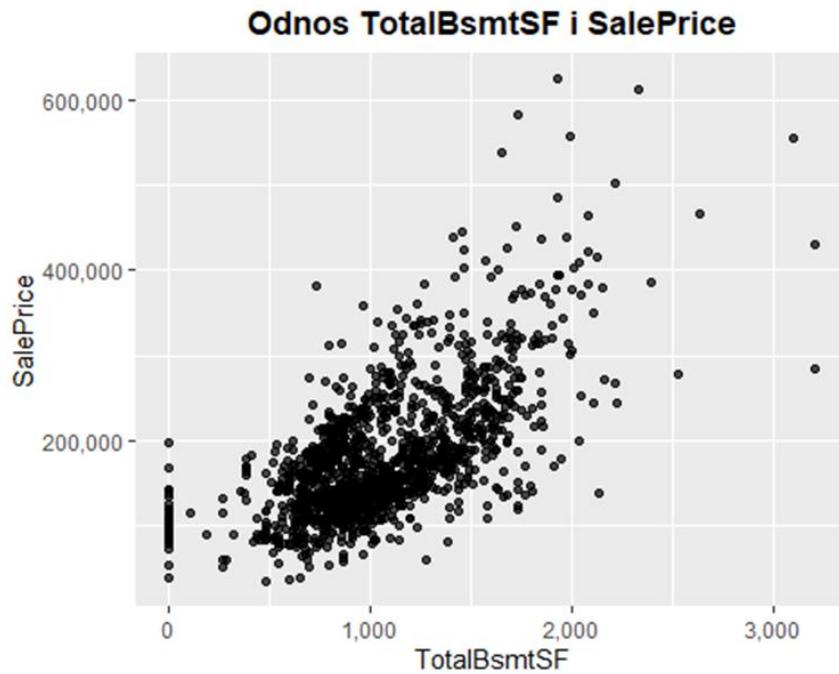
- Veličina kuće: veće kuće obično imaju veću cenu, pa su važni atributi GrLivArea, TotalBsmtSF i LotArea.
- Starost: novije kuće obično vrede više, relevantni atribut je YearBuilt.
- Kvalitet: kvalitet izgradnje i enterijera utiče na cenu, pa se prate OverallQual, OverallCond i KitchenQual.
- Lokacija: lokacija značajno utiče na cenu, stoga je Neighborhood ključni atribut.
- Pogodnosti: prisustvo garaže, kamina i broja kupatila takođe utiče na vrednost, što obuhvata GarageCars, Fireplaces i FullBath.

```
ggplot(data.train, aes(x = GrLivArea, y = SalePrice)) +
  geom_point(alpha = 0.7) +
  scale_x_continuous(labels = comma) +
  scale_y_continuous(labels = comma) +
  labs(
    title = "Odnos GrLivArea i SalePrice",
    x = "GrLivArea",
    y = "SalePrice"
  ) +
  theme(plot.title = element_text(hjust = 0.5, face = "bold"))
```



Grafik pokazuje pozitivnu korelaciju - kako se površina stambenog prostora povećava, tako raste i cena kuće, što ovu promenljivu čini pogodnom za model.

```
ggplot(data.train, aes(x = TotalBsmtSF, y = SalePrice)) +
  geom_point(alpha = 0.7) +
  scale_x_continuous(labels = comma) +
  scale_y_continuous(labels = comma) +
  labs(
    title = "Odnos TotalBsmtSF i SalePrice",
    x = "TotalBsmtSF",
    y = "SalePrice"
  ) +
  theme(plot.title = element_text(hjust = 0.5, face = "bold"))
```

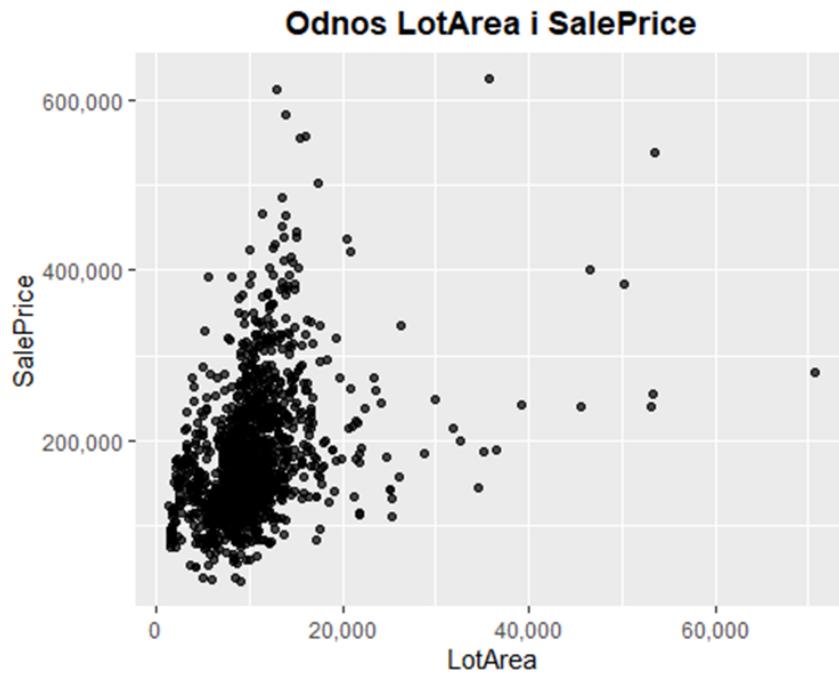


Grafik prikazuje vezu između ukupne kvadrature podruma i cene kuće.

Primećuje se pozitivna korelacija - veći podrumi su povezani sa većom cenom kuće.

Ipak, odnos nije savršeno linearan: kod većih kvadratura dolazi do rasipanja tačaka, gde neke kuće imaju nižu cenu nego što je očekivano.

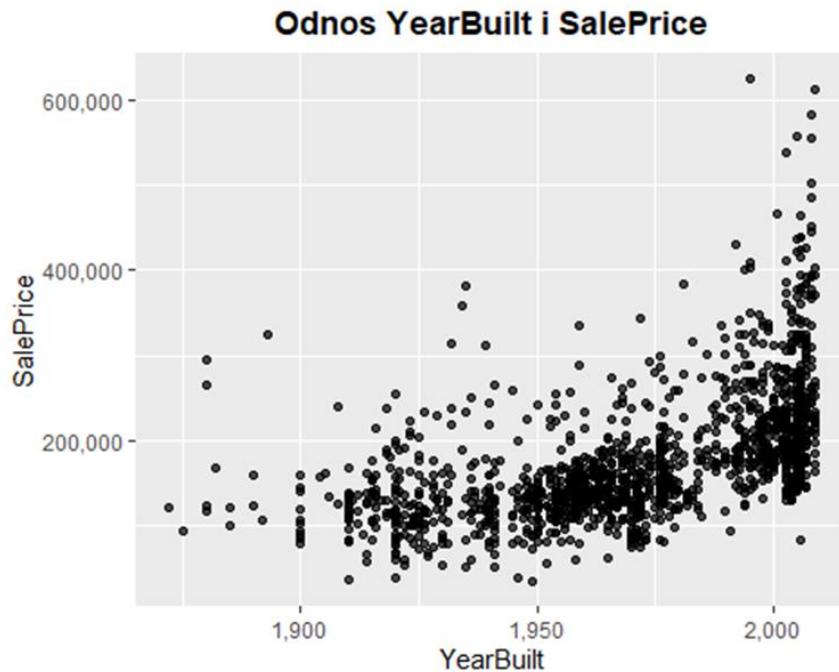
```
ggplot(data.train, aes(x = LotArea, y = SalePrice)) +
  geom_point(alpha = 0.7) +
  scale_x_continuous(labels = comma) +
  scale_y_continuous(labels = comma) +
  labs(
    title = "Odnos LotArea i SalePrice",
    x = "LotArea",
    y = "SalePrice"
  ) +
  theme(plot.title = element_text(hjust = 0.5, face = "bold"))
```



Grafik prikazuje odnos između veličine parcele i cene kuće.

Može se uočiti blaga pozitivna korelacija - kuće sa većim placem generalno imaju veću cenu, ali ta veza nije jaka niti linearna.

```
ggplot(data.train, aes(x = YearBuilt, y = SalePrice)) +
  geom_point(alpha = 0.7) +
  scale_x_continuous(labels = comma) +
  scale_y_continuous(labels = comma) +
  labs(
    title = "Odnos YearBuilt i SalePrice",
    x = "YearBuilt",
    y = "SalePrice"
  ) +
  theme(plot.title = element_text(hjust = 0.5, face = "bold"))
```



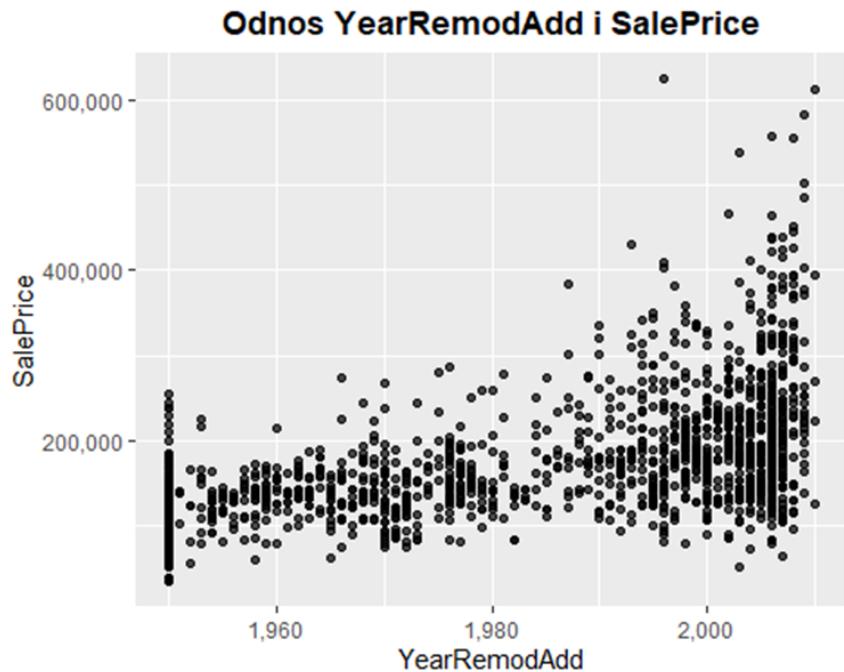
Grafik prikazuje odnos između godine izgradnje kuće i njene cene.

Novije kuće (izgrađene posle 1980.) generalno imaju veće cene u poređenju sa starijim.

Kod kuća izgrađenih pre 1950. godine cene su uglavnom niže, ali postoji nekoliko izuzetaka, verovatno zbog renoviranja ili lokacije.

Od 2000. godine, može se videti veći broj skupljih kuća.

```
ggplot(data.train, aes(x = YearRemodAdd, y = SalePrice)) +
  geom_point(alpha = 0.7) +
  scale_x_continuous(labels = comma) +
  scale_y_continuous(labels = comma) +
  labs(
    title = "Odnos YearRemodAdd i SalePrice",
    x = "YearRemodAdd",
    y = "SalePrice"
  ) +
  theme(plot.title = element_text(hjust = 0.5, face = "bold"))
```

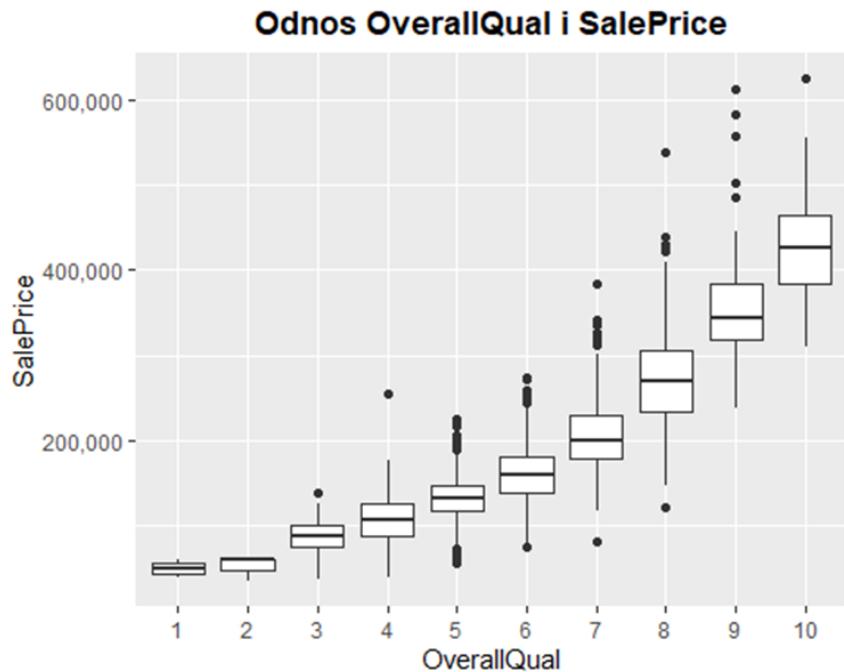


Grafik prikazuje vezu između godine poslednjeg renoviranja kuće i cene.

Primećuje se blaga pozitivna korelacija — kuće koje su nedavno renovirane (posebno posle 2000. godine) imaju veće cene.

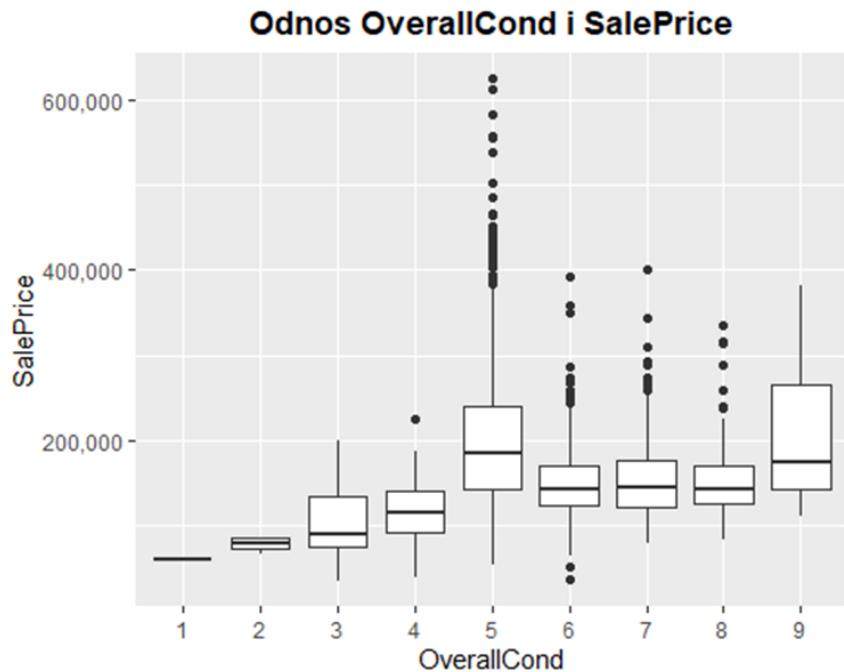
Starije kuće bez skorijih renoviranja generalno imaju niže cene, što sugerije da renoviranje povećava tržišnu vrednost.

```
ggplot(data.train, aes(x = as.factor(OverallQual), y = SalePrice)) +
  geom_boxplot() +
  scale_y_continuous(labels = comma) +
  labs(
    title = "Odnos OverallQual i SalePrice",
    x = "OverallQual",
    y = "SalePrice"
  ) +
  theme(plot.title = element_text(hjust = 0.5, face = "bold"))
```



Ovaj boxplot prikazuje raspodelu cene kuće u zavisnosti od ukupnog kvaliteta kuće. Jasno se vidi pozitivna korelacija - kako se kvalitet kuće povećava, tako raste i medijana cene.

```
ggplot(data.train, aes(x = as.factor(OverallCond), y = SalePrice)) +
  geom_boxplot() +
  scale_y_continuous(labels = comma) +
  labs(
    title = "Odnos OverallCond i SalePrice",
    x = "OverallCond",
    y = "SalePrice"
  ) +
  theme(plot.title = element_text(hjust = 0.5, face = "bold"))
```



Ovaj grafik pokazuje kako ocena stanja kuće utiče na cenu. Cena kuće se ne menja značajno sa promenom OverallCond, što znači da stanje kuće nije toliko snažan prediktor cene.

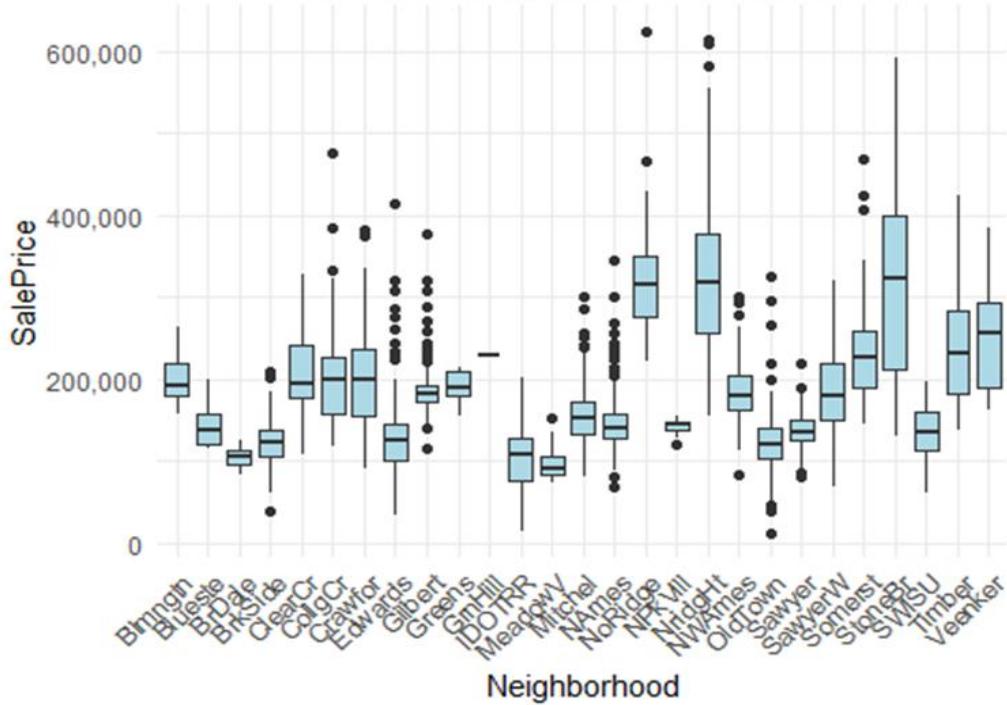
```
ggplot(data.train, aes(x = as.factor(KitchenQual), y = SalePrice)) +
  geom_boxplot() +
  scale_y_continuous(labels = comma) +
  labs(
    title = "Odnos KitchenQual i SalePrice",
    x = "KitchenQual",
    y = "SalePrice"
  ) +
  theme(plot.title = element_text(hjust = 0.5, face = "bold"))
```



Ovaj boxplot pokazuje uticaj kvaliteta kuhinje na cenu kuće. Kuće sa boljim kvalitetom kuhinje generalno imaju veće cene, dok kuće sa kuhinjom nižeg kvaliteta imaju niže cene.

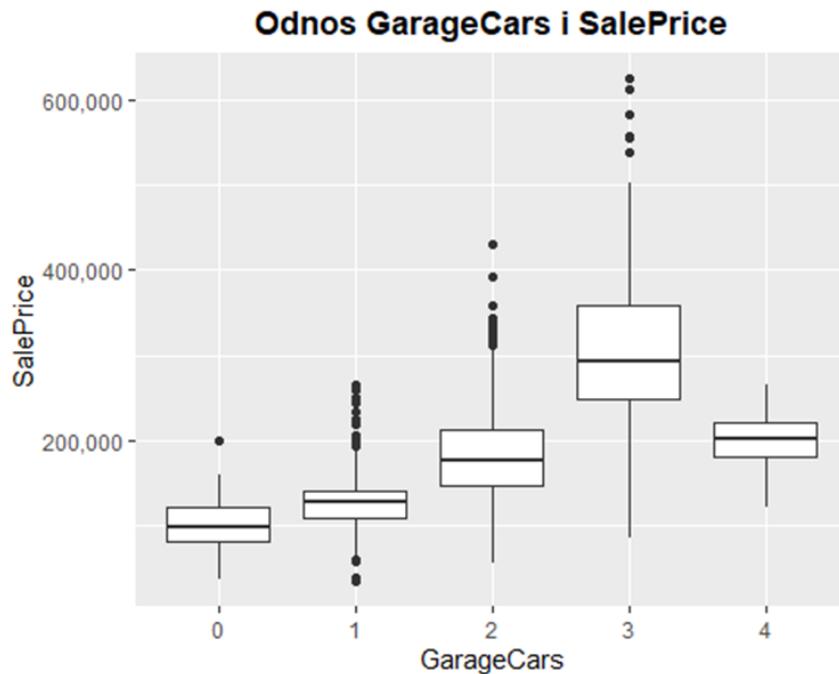
```
ggplot(data.train, aes(x = as.factor(Neighborhood), y = SalePrice)) +
  geom_boxplot() +
  scale_y_continuous(labels = comma) +
  labs(
    title = "Odnos Neighborhood i SalePrice",
    x = "Neighborhood",
    y = "SalePrice"
  ) +
  theme(plot.title = element_text(hjust = 0.5, face = "bold"))
```

## Odnos Neighborhood i SalePrice



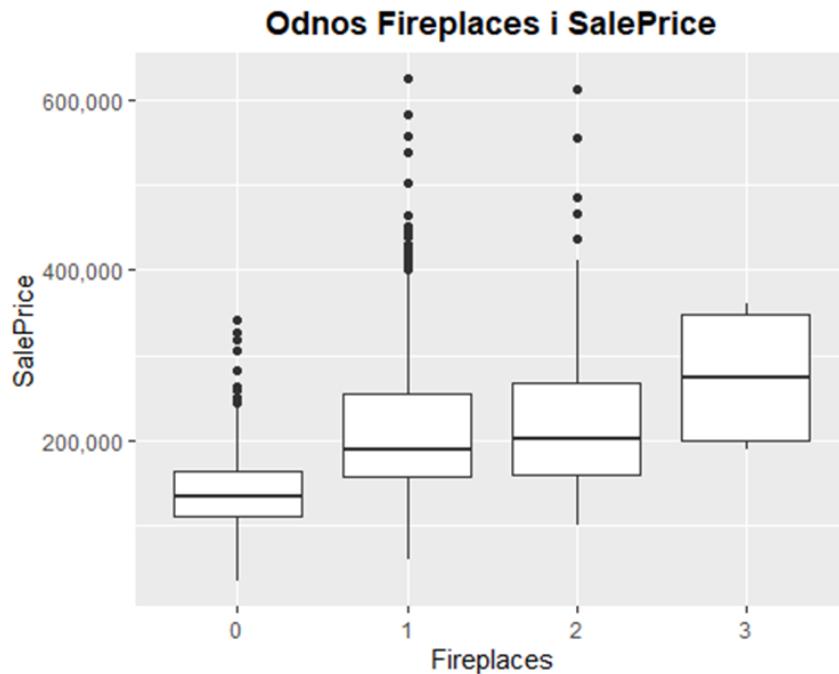
Ovaj boxplot prikazuje raspodelu cena kuća po različitim kvartovima. Jasno se vidi da lokacija značajno utiče na cenu. Kvartovi kao što su NoRidge, NridgHt i StoneBr imaju znatno više medijane cene, dok kvartovi poput MeadowV, BrDale i IDOTRR imaju niže medijane.

```
ggplot(data.train, aes(x = as.factor(GarageCars), y = SalePrice)) +
  geom_boxplot() +
  scale_y_continuous(labels = comma) +
  labs(
    title = "Odnos GarageCars i SalePrice",
    x = "GarageCars",
    y = "SalePrice"
  ) +
  theme(plot.title = element_text(hjust = 0.5, face = "bold"))
```



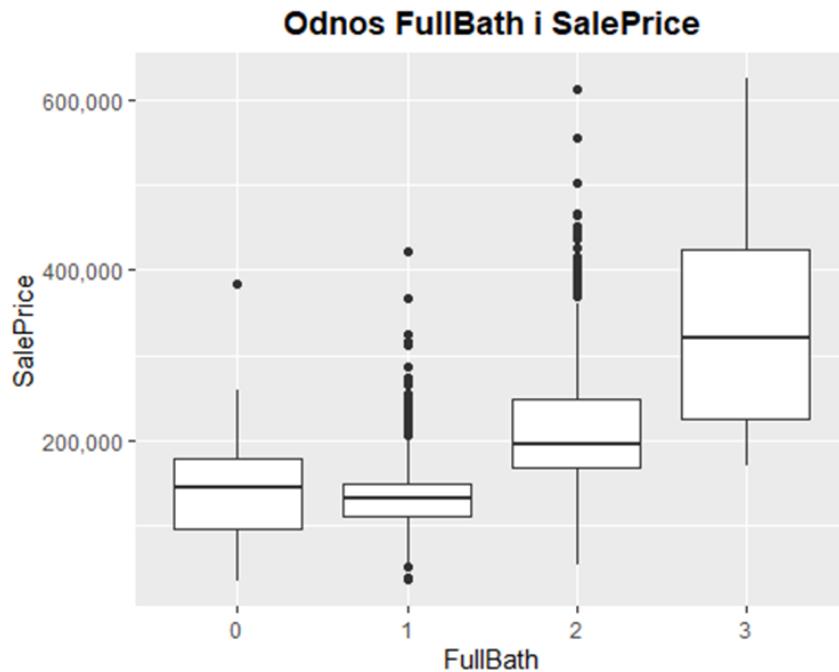
Ovaj boxplot prikazuje kako broj parking mesta u garaži utiče na cenu kuće. Veći broj parking mesta je generalno u korelaciji sa višim cenama - kuće sa 3 ili 4 mesta u garaži imaju znatno višu medijanu cenu od kuća sa 0,1 mestom.

```
ggplot(data.train, aes(x = as.factor(Fireplaces), y = SalePrice)) +
  geom_boxplot() +
  scale_y_continuous(labels = comma) +
  labs(
    title = "Odnos Fireplaces i SalePrice",
    x = "Fireplaces",
    y = "SalePrice"
  ) +
  theme(plot.title = element_text(hjust = 0.5, face = "bold"))
```



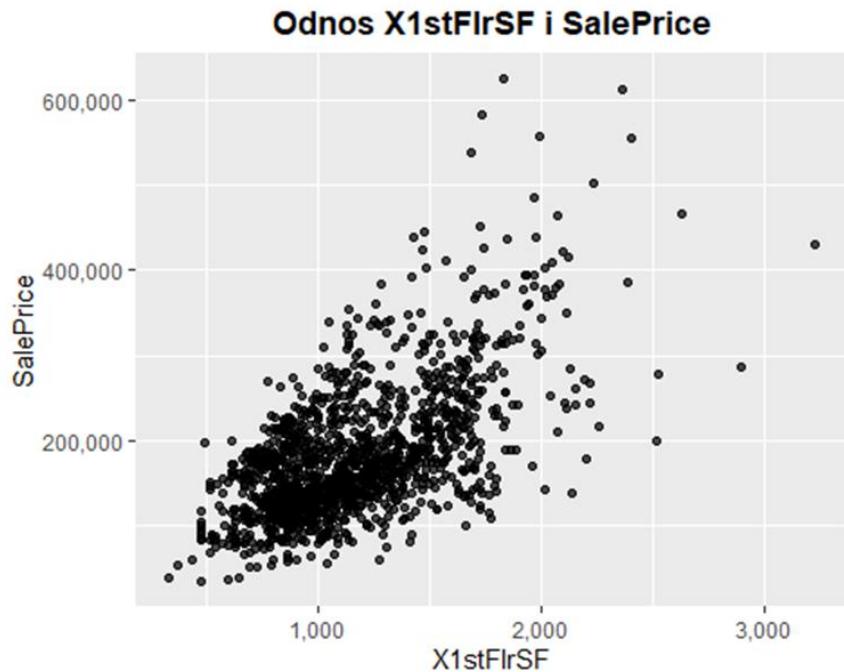
Ovaj boxplot prikazuje kako broj kamina utiče na cenu kuće. Kuće sa većim brojem kamina generalno imaju višu medijanu cenu.

```
ggplot(data.train, aes(x = as.factor(FullBath), y = SalePrice)) +
  geom_boxplot() +
  scale_y_continuous(labels = comma) +
  labs(
    title = "Odnos FullBath i SalePrice",
    x = "FullBath",
    y = "SalePrice"
  ) +
  theme(plot.title = element_text(hjust = 0.5, face = "bold"))
```



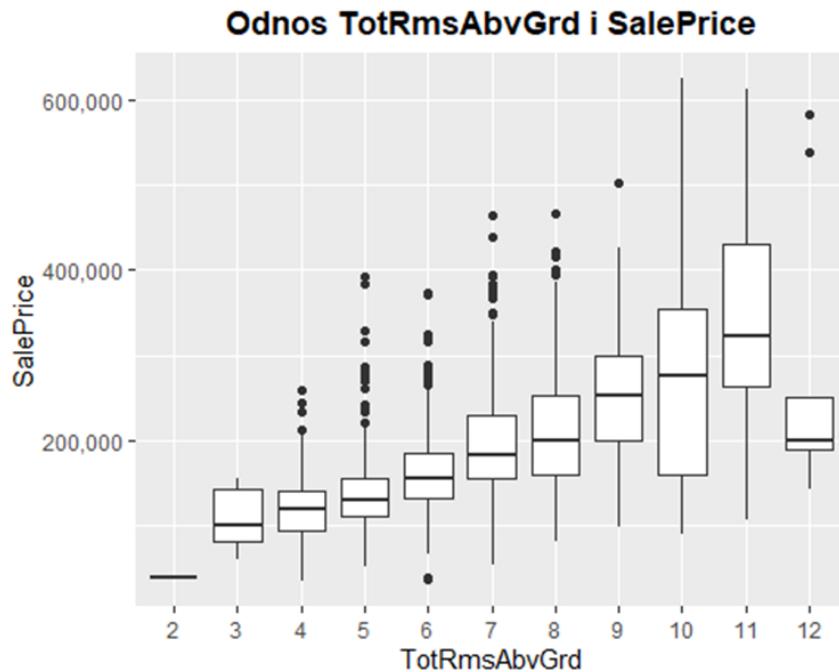
Ovaj grafikon prikazuje odnos između broja kupatila i prodajne cene kuće. Primećuje se rast cena sa povećanjem broja kupatila - kuće sa više kupatila imaju veću cenu.

```
ggplot(data.train, aes(x = X1stFlrSF, y = SalePrice)) +
  geom_point(alpha = 0.7) +
  scale_x_continuous(labels = comma) +
  scale_y_continuous(labels = comma) +
  labs(
    title = "Odnos X1stFlrSF i SalePrice",
    x = "X1stFlrSF",
    y = "SalePrice"
  ) +
  theme(plot.title = element_text(hjust = 0.5, face = "bold"))
```



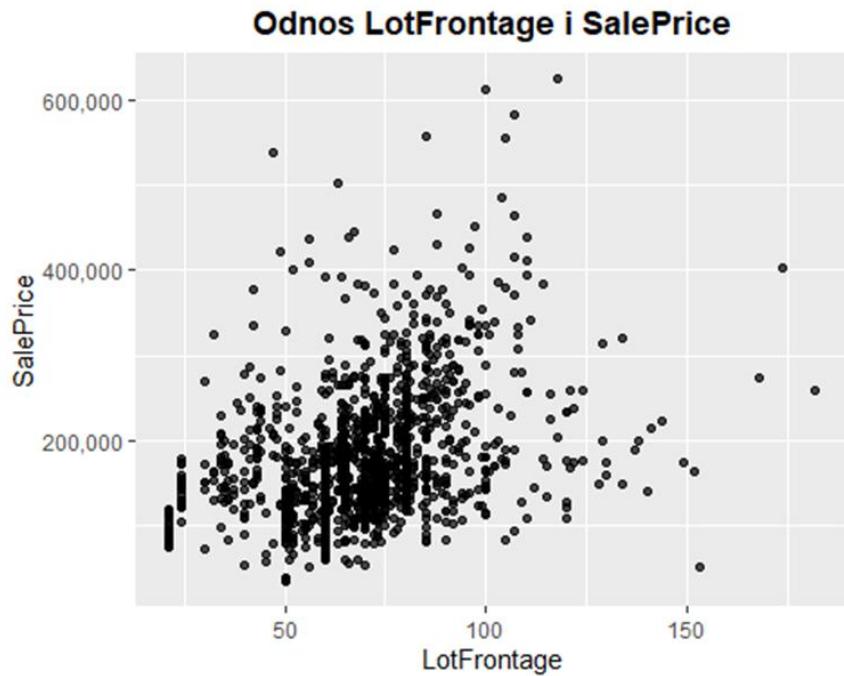
Ovaj grafikon prikazuje odnos između površine prvog sprata i cene kuće. Postoji pozitivna korelacija - kuće sa većom površinom prvog sprata uglavnom imaju veću cenu.

```
ggplot(data.train, aes(x = as.factor(TotRmsAbvGrd), y = SalePrice)) +
  geom_boxplot() +
  scale_y_continuous(labels = comma) +
  labs(
    title = "Odnos TotRmsAbvGrd i SalePrice",
    x = "TotRmsAbvGrd",
    y = "SalePrice"
  ) +
  theme(plot.title = element_text(hjust = 0.5, face = "bold"))
```



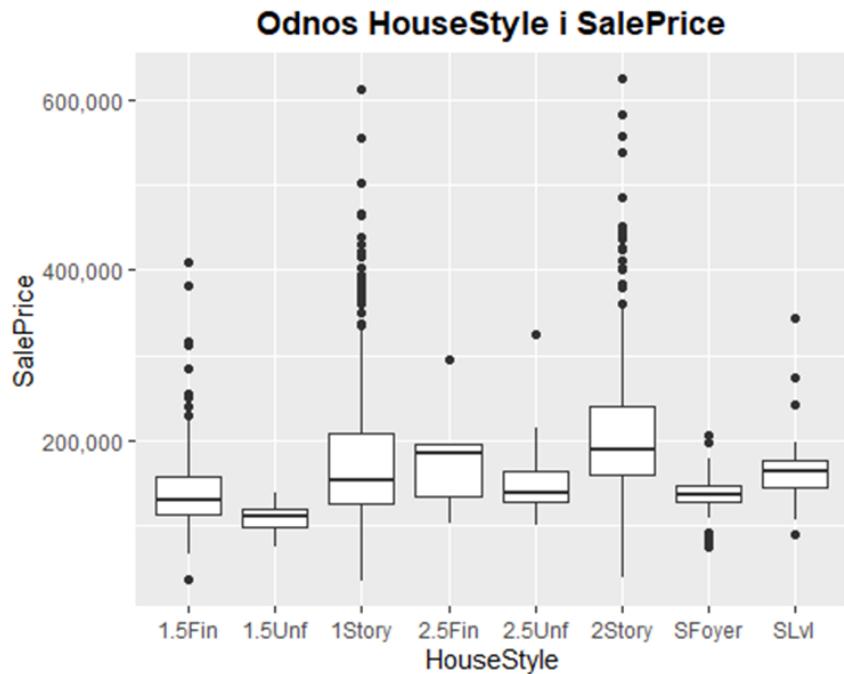
Ovaj grafikon prikazuje odnos između ukupnog broja soba iznad zemlje i prodajne cene kuće. Veći broj soba utiče na porast cene kuće.

```
ggplot(data.train, aes(x = LotFrontage, y = SalePrice)) +
  geom_point(alpha = 0.7) +
  scale_x_continuous(labels = comma) +
  scale_y_continuous(labels = comma) +
  labs(
    title = "Odnos LotFrontage i SalePrice",
    x = "LotFrontage",
    y = "SalePrice"
  ) +
  theme(plot.title = element_text(hjust = 0.5, face = "bold"))
```



Ovaj grafikon prikazuje odnos između dužine prilaznog puta i cene kuće. Kuće sa većim LotFrontage generalno imaju tendenciju da budu skuplje, iako veza nije potpuno linearna.

```
ggplot(data.train, aes(x = as.factor(HouseStyle), y = SalePrice)) +
  geom_boxplot() +
  scale_y_continuous(labels = comma) +
  labs(
    title = "Odnos HouseStyle i SalePrice",
    x = "HouseStyle",
    y = "SalePrice"
  ) +
  theme(plot.title = element_text(hjust = 0.5, face = "bold"))
```



Ovaj boxplot prikazuje raspodelu cena kuća u zavisnosti od stila kuće. Različiti stilovi kuća pokazuju različite medijane i raspon cena. Neki stilovi (2Story, 2.5Fin) imaju tendenciju ka višim cenama.

```
ggplot(data.train, aes(x = as.factor(OverallQual), y = SalePrice)) +
  geom_boxplot() +
  scale_y_continuous(labels = comma) +
  labs(
    title = "Odnos OverallQual i SalePrice",
    x = "OverallQual",
    y = "SalePrice"
  ) +
  theme(plot.title = element_text(hjust = 0.5, face = "bold"))
```

# Izbor promenljivih / Feature Engineering

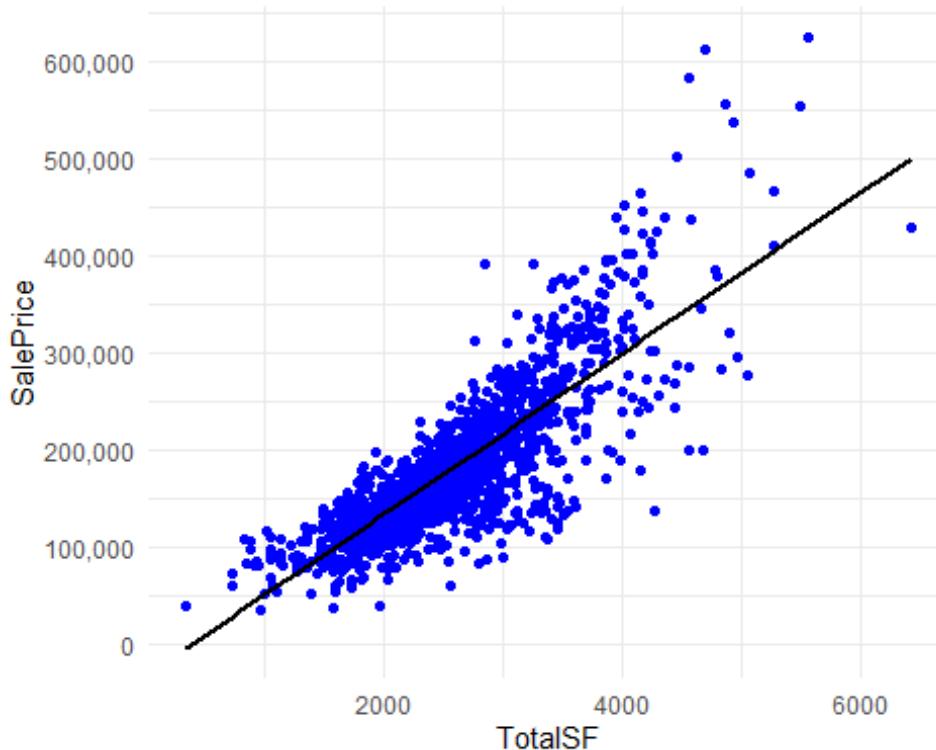
## Feature Engineering

Kreirana je promenljiva TotalSF, koja predstavlja ukupnu površinu kuće. Analiza pokazuje jaku pozitivnu korelaciju između ukupne površine i prodajne cene.

```
data <- rbind(data.train, data.test)

data$TotalSF <- data$TotalBsmtSF + data$GrLivArea

ggplot(data = data[!is.na(data$SalePrice),], aes(x = TotalSF, y = SalePrice))
+
  geom_point(col = 'blue') +
  geom_smooth(method = "lm", se = FALSE, color = "black") +
  scale_y_continuous(breaks = seq(0, 800000, by = 100000), labels = scales::comma) +
  theme_minimal()
## `geom_smooth()` using formula = 'y ~ x'
```



Dodata je promenljiva HouseAge, koja predstavlja starost kuće u trenutku prodaje. Starost kuće je pogodnija za predikciju cene od same godine izgradnje. Analiza pokazuje da starije kuće imaju tendenciju da budu jeftinije.

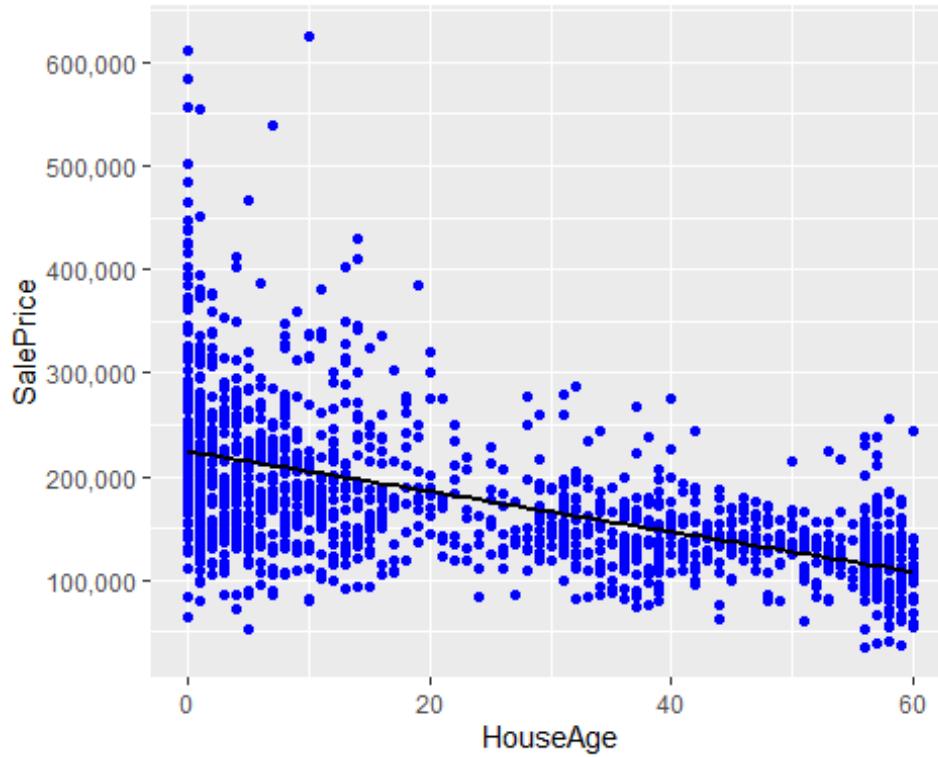
Takođe je dodata promenljiva HouseRemodAge - godina poslednjeg renoviranja kuće.

```
data$HouseAge <- data$YrSold - data$YearBuilt

data$HouseRemodAge <- data$YrSold - data$YearRemodAdd

ggplot(data=data[ !is.na(data$SalePrice),], aes(x=HouseAge, y=SalePrice))+
  geom_point(col='blue') + geom_smooth(method = "lm", se=FALSE, color="black",
  aes(group=1)) +
  scale_y_continuous(breaks= seq(0, 800000, by=100000), labels = comma)

## `geom_smooth()` using formula = 'y ~ x'
```



Kreirana je i promenljiva TotalFinishedSF, koja predstavlja ukupnu završenu površinu kuće.

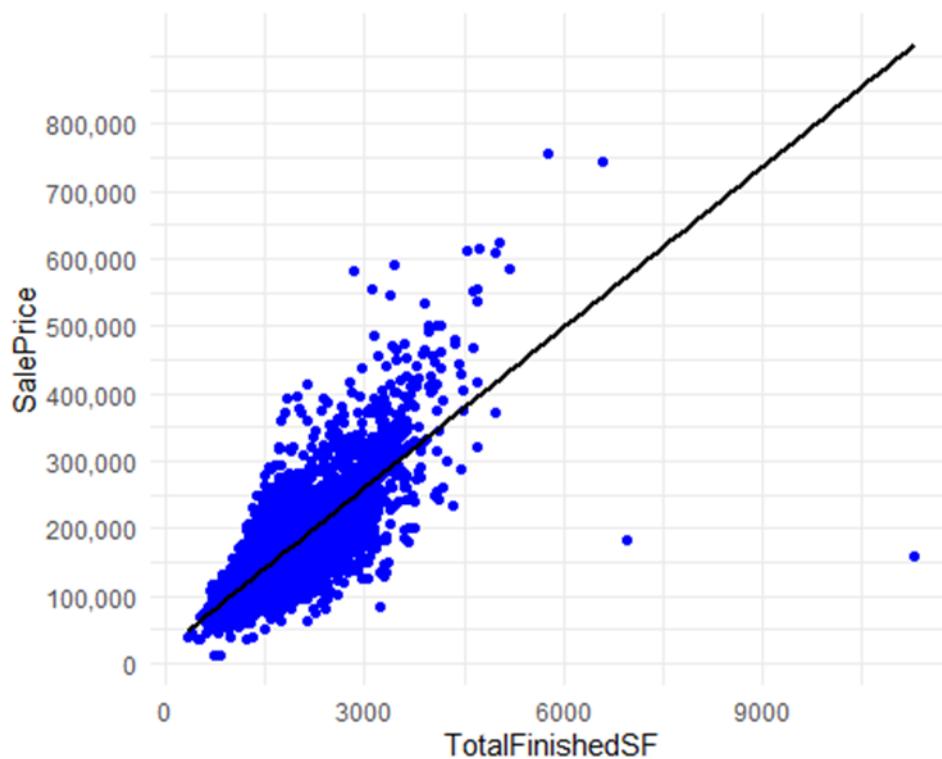
```

data$TotalFinishedSF <- data$GrLivArea + data$BsmtFinSF1 + data$BsmtFinSF2

ggplot(data = data[!is.na(data$SalePrice),], aes(x = TotalFinishedSF, y = SalePrice)) +
  geom_point(col = 'blue') +
  geom_smooth(method = "lm", se = FALSE, color = "black") +
  scale_y_continuous(breaks = seq(0, 800000, by = 100000), labels = scales::comma) +
  theme_minimal()

## `geom_smooth()` using formula = 'y ~ x'

```



Dodate su i binarne promenljive IsNew - da li je kuća nova, HasBsmt - da li ima podrum, HasBath - da li ima kupatilo, HasGarage - da li kuća ima garažu, HasFireplace - da li kuća ima kamin, HasPool - da li kuća ima bazen.

```

data$IsNew <- ifelse(data$YrSold==data$YearBuilt, 1, 0)
data$IsNew = as.factor(data$IsNew)

data$HasBath <- ifelse(data$BsmtFullBath + data$FullBath + 0.5 * (data$BsmtHalfBath + data$HalfBath) > 0, 1, 0)

```

```

data$HasBath <- as.factor(data$HasBath)

data$HasBsmt <- ifelse(data$TotalBsmtSF > 0, 1, 0)
data$HasBsmt <- as.factor(data$HasBsmt)

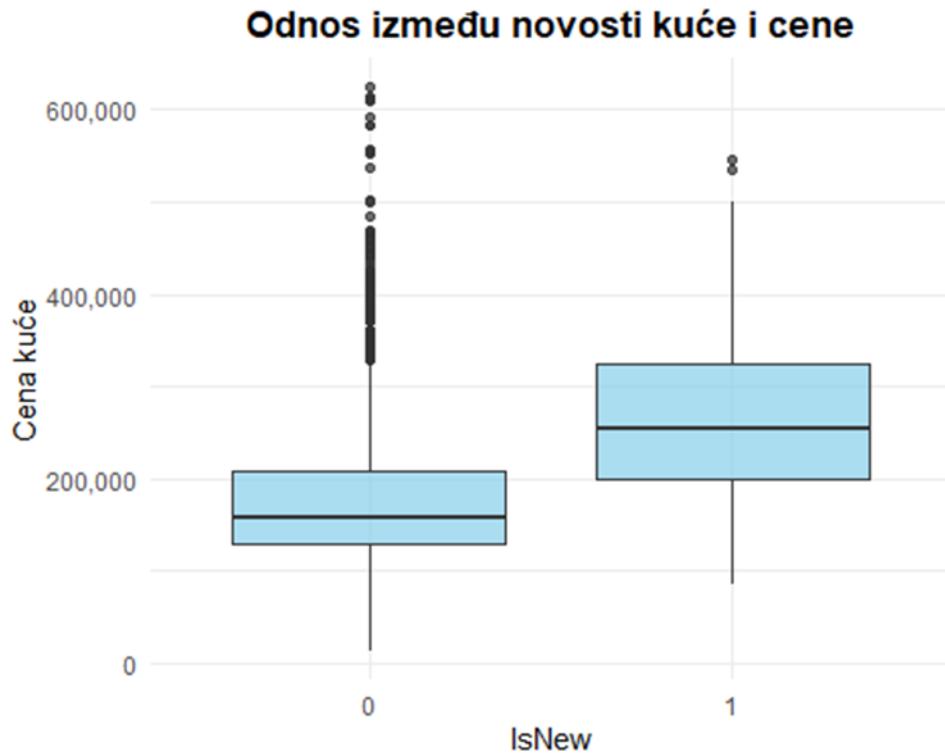
# garage, fireplace, pool
data$HasGarage <- ifelse(data$GarageArea > 0, 1, 0)
data$HasGarage <- as.factor(data$HasGarage)

data$HasFireplace <- ifelse(data$Fireplaces > 0, 1, 0)
data$HasFireplace <- as.factor(data$HasFireplace)

data$HasPool <- ifelse(data$PoolArea > 0, 1, 0)
data$HasPool <- as.factor(data$HasPool)

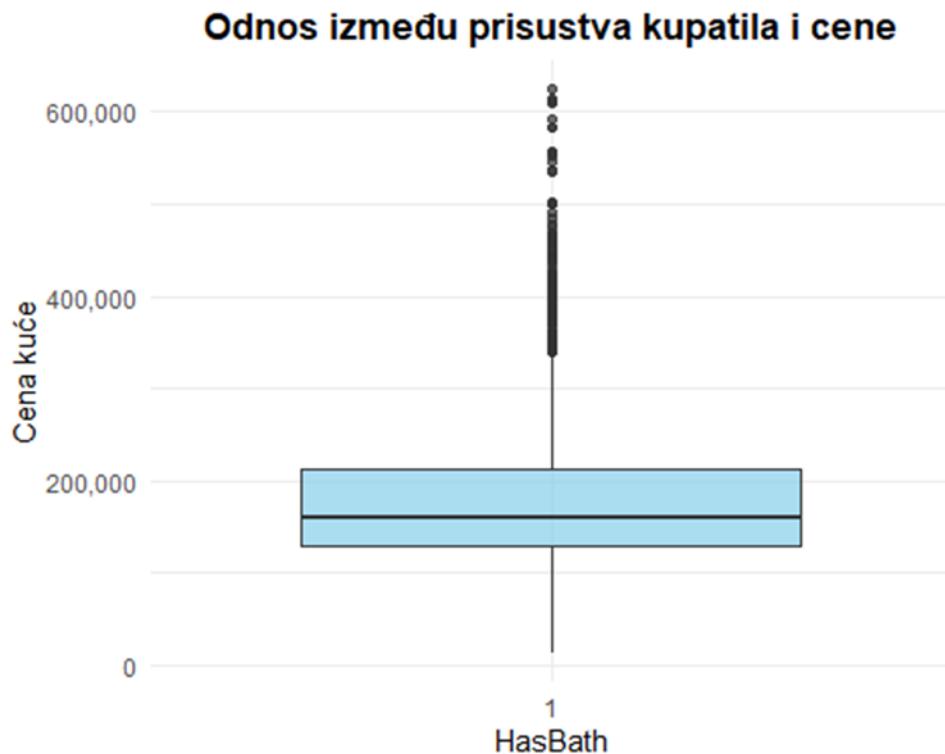
ggplot(data, aes(x = IsNew, y = SalePrice)) +
  geom_boxplot(fill = "skyblue", alpha = 0.7) +
  scale_y_continuous(labels = comma) +
  labs(title = "Odnos između novosti kuće i cene",
       x = "IsNew",
       y = "Cena kuće") +
  theme_minimal() +
  theme(plot.title = element_text(hjust = 0.5, face = "bold"))

```



Cena zavisi od toga da li je kuća nova.

```
ggplot(data, aes(x = HasBath, y = SalePrice)) +
  geom_boxplot(fill = "skyblue", alpha = 0.7) +
  scale_y_continuous(labels = comma) +
  labs(title = "Odnos između prisustva kupatila i cene",
       x = "HasBath",
       y = "Cena kuće") +
  theme_minimal() +
  theme(plot.title = element_text(hjust = 0.5, face = "bold"))
```



Svaka kuća ima makar jedno kupatilo, pa nam ovaj atribut nije značajan.

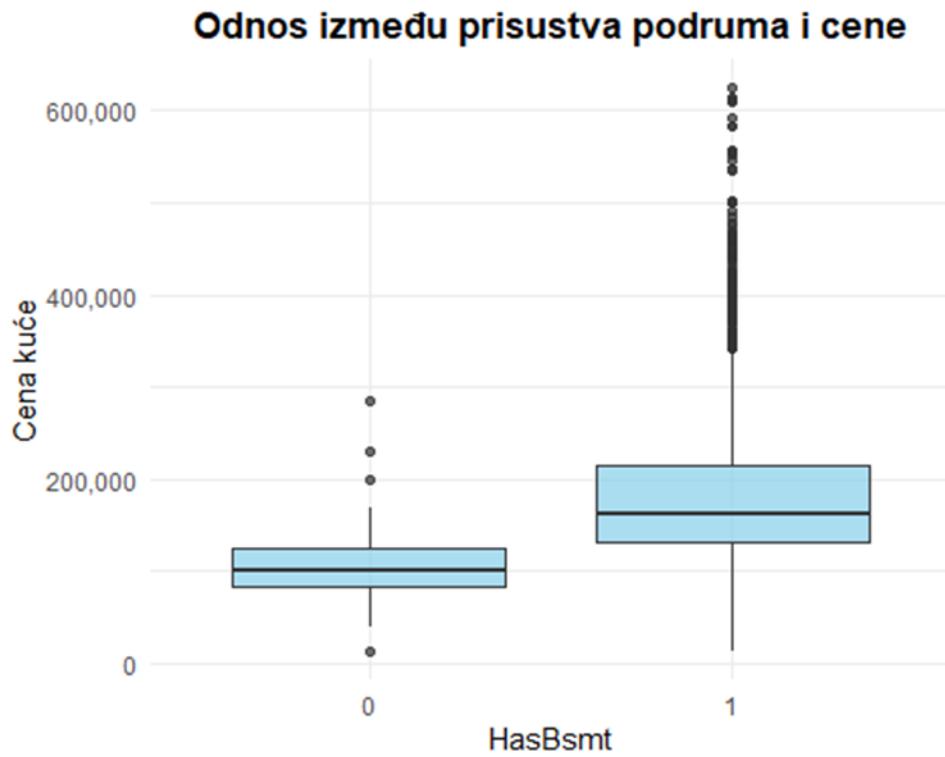
```
data$HasBath <- NULL

ggplot(data, aes(x = HasBsmt, y = SalePrice)) +
  geom_boxplot(fill = "skyblue", alpha = 0.7) +
  scale_y_continuous(labels = comma) +
  labs(title = "Odnos između prisustva podruma i cene",
       x = "HasBsmt",
```

```

y = "Cena kuce") +
theme_minimal() +
theme(plot.title = element_text(hjust = 0.5, face = "bold"))

```

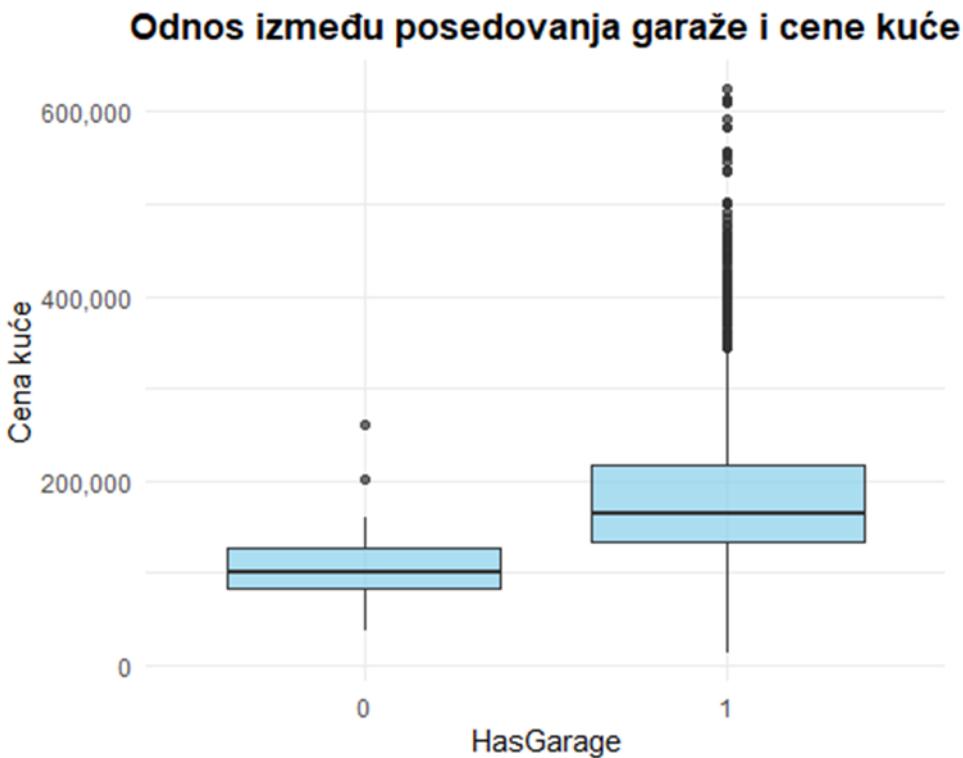


Prisustvo podruma utiče na SalePrice.

```

ggplot(data, aes(x = HasGarage, y = SalePrice)) +
  geom_boxplot(fill = "skyblue", alpha = 0.7) +
  scale_y_continuous(labels = scales::comma) +
  labs(
    title = "Odnos izmedju posedovanja garaze i cene kuće",
    x = "HasGarage",
    y = "Cena kuće"
  ) +
  theme_minimal() +
  theme(plot.title = element_text(hjust = 0.5, face = "bold"))

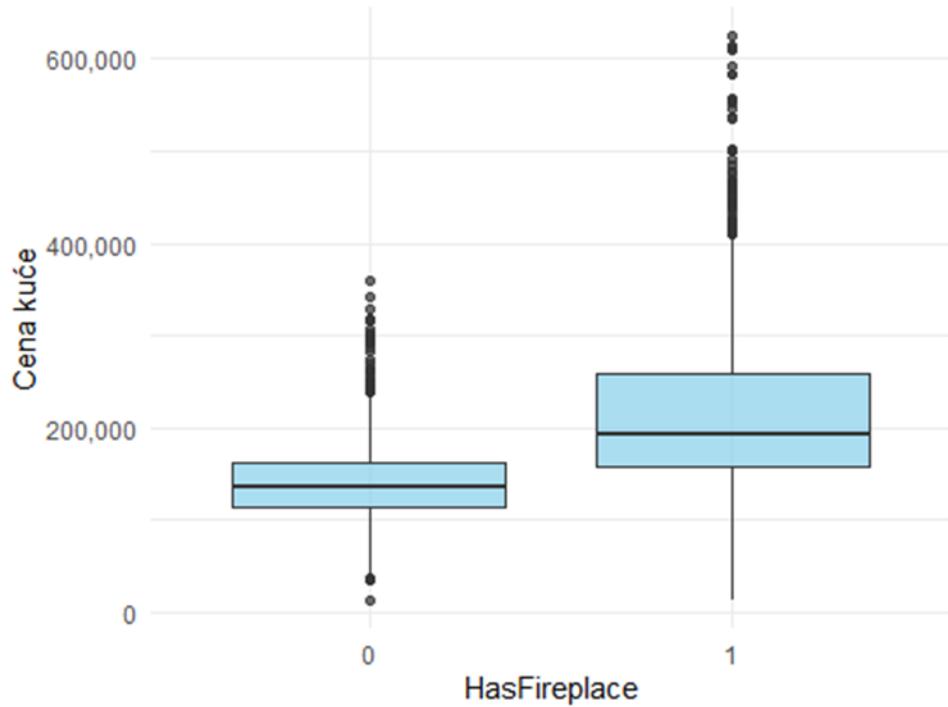
```



Postojanje garaže utiče na SalePrice.

```
ggplot(data, aes(x = HasGarage, y = SalePrice)) +
  geom_boxplot(fill = "skyblue", alpha = 0.7) +
  scale_y_continuous(labels = scales::comma) +
  labs(
    title = "Odnos izmedju posedovanja garaže i cene kuće",
    x = "HasGarage",
    y = "Cena kuće"
  ) +
  theme_minimal() +
  theme(plot.title = element_text(hjust = 0.5, face = "bold"))
```

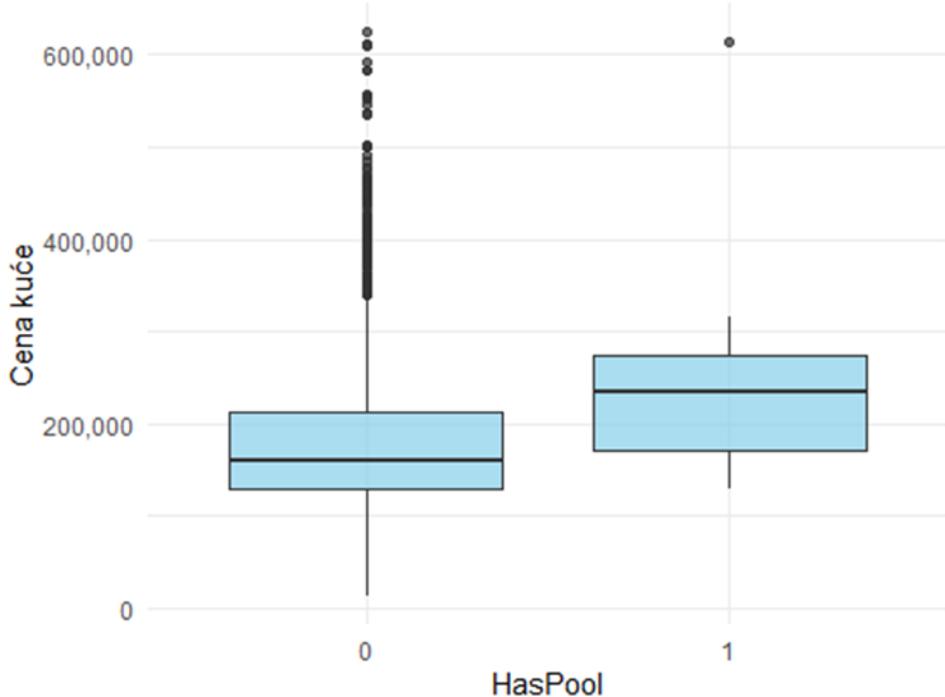
### Odnos između posedovanja kamina i cene kuće



Postojanje kamina utiče na SalePrice.

```
ggplot(data, aes(x = HasFireplace, y = SalePrice)) +
  geom_boxplot(fill = "skyblue", alpha = 0.7) +
  scale_y_continuous(labels = scales::comma) +
  labs(
    title = "Odnos izmedju posedovanja bazena i cene kuće",
    x = "HasFireplace",
    y = "Cena kuće"
  ) +
  theme_minimal() +
  theme(plot.title = element_text(hjust = 0.5, face = "bold"))
```

## Odnos između posedovanja bazena i cene kuće



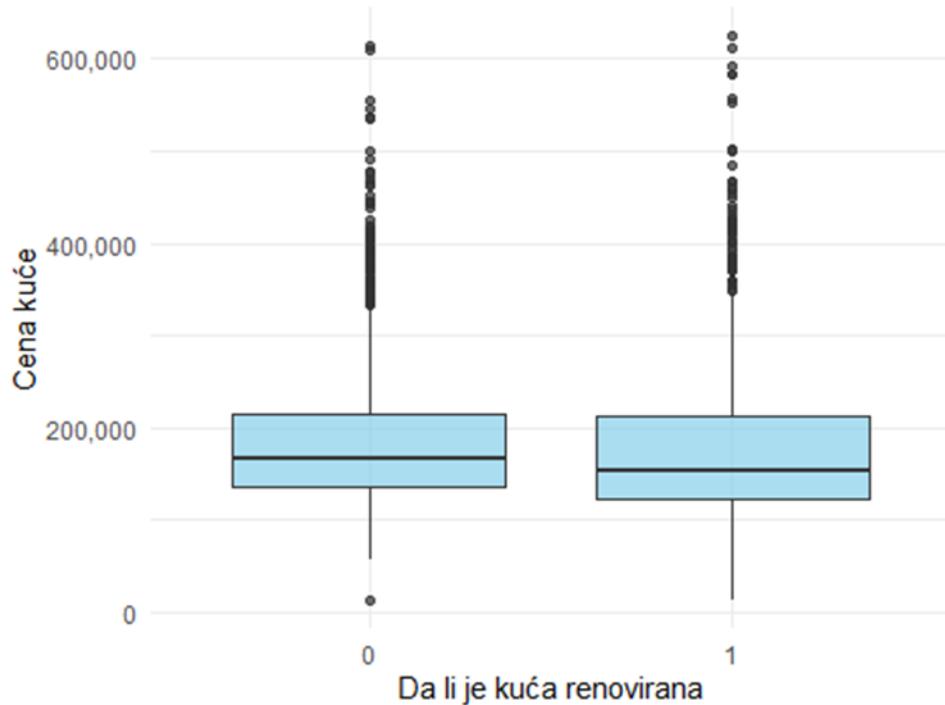
Postojanje bazena utiče na SalePrice, ali malo kuća poseduje bazen.

Uvedena je binarna promenljiva Remodeled, koja označava da li je kuća renovirana ili nije. Analiza pokazuje da kuće koje nisu renovirane imaju malo veću medijanu cenu.

```
data$Remodeled <- ifelse(data$YearBuilt==data$YearRemodAdd, 0, 1)
# 0 - ne
# 1 - da
data$Remodeled <- as.factor(data$Remodeled)

ggplot(data=data[!is.na(data$SalePrice),], aes(x = Remodeled, y = SalePrice))
+
  geom_boxplot(fill = "skyblue", alpha = 0.7) +
  scale_y_continuous(labels = scales::comma) +
  labs(
    title = "Odnos između renoviranja i cene kuće",
    x = "Da li je kuća renovirana",
    y = "Cena kuće (SalePrice)"
  ) +
  theme_minimal() +
  theme(
    plot.title = element_text(hjust = 0.5, face = "bold"),
  )
```

### Odnos između renoviranja i cene kuće



Dodavanjem promenljive TotalPorchSF, koja predstavlja ukupnu površinu svih terasa i verandi, ispitana je njena veza sa prodajnom cenom. Korelacija sa ciljnom promenljivom se pokazala kao slaba.

```
data$TotalPorchSF <- data$OpenPorchSF + data$EnclosedPorch +
  data$X3SsnPorch + data$ScreenPorch
cor(data$SalePrice, data$TotalPorchSF, use= "pairwise.complete.obs")
## [1] 0.209169
```

Dodata je promenljiva TotalBaths, koja označava ukupan broj kupatila. Postoji određena korelacija između ciljne promenljive i TotalBaths.

Dodata je promenljiva TotalBaths, koja označava ukupan broj kupatila. Od nje ćemo napraviti ordinalnu, kategoriju promenljivu sa nivoima: malo (< 2), srednje (2,3), mnogo (> 3).

Postoji korelacija između ciljne promenljive i TotalBaths.

```

data$TotalBaths <- data$BsmtFullBath + data$FullBath +
  0.5 * (data$BsmtHalfBath + data$HalfBath)

data$TotalBaths <- cut(
  data$TotalBaths,
  breaks = c(-Inf, 1.5, 3, Inf),
  labels = c("Malo", "Srednje", "Mnogo"),
  right = TRUE
)

table(data$TotalBaths)

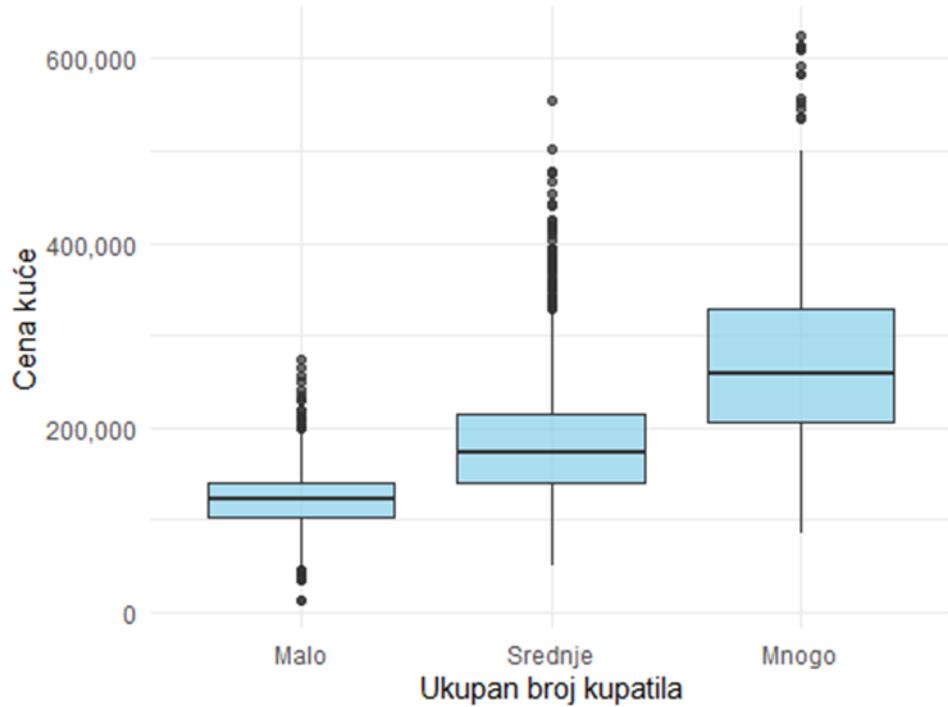
##
##      Malo Srednje   Mnogo
##      729     1832     342

ggplot(data=data, aes(x = TotalBaths, y = SalePrice)) +
  geom_boxplot(fill = "skyblue", alpha = 0.7) +
  scale_y_continuous(labels = scales::comma) +
  labs(
    title = "Odnos izmedu ukupnog broja kupatila i cene kuce",
    x = "Ukupan broj kupatila",
    y = "Cena kuce (SalePrice)"
  ) +
  theme_minimal() +
  theme(
    plot.title = element_text(hjust = 0.5, face = "bold"),
  )

data$TotalBaths <- as.integer(data$TotalBaths)

```

### Odnos između ukupnog broja kupatila i cene kuć



Izbor promenljivih za model zasnovan je na rezultatima analize korelacije sprovedene tokom EDA faze. Izdvojene su promenljive koje pokazuju najjaču povezanost sa ciljnom promenljivom SalePrice, dok su one sa slabom ili zanemarljivom korelacijom isključene. Ovakav pristup omogućava smanjenje dimenzionalnosti i poboljšanje efikasnosti modela, uz zadržavanje najrelevantnijih prediktora.

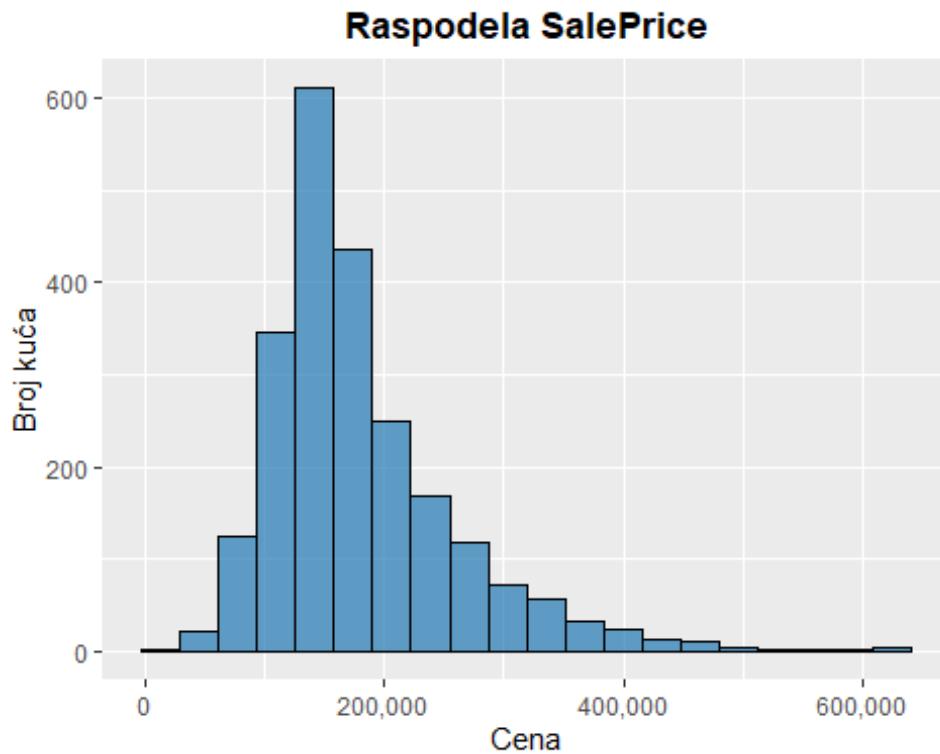
# Implementacija i procena modela

Cilj ovog seminarskog rada je kreiranje robustnih modela koji će moći da predvide cenu kuće (SalePrice) u zavisnosti od vrednosti ostalih atributa koji opisuju svaku kuću. Predviđanje cene, koja je kontinuirani tip podatka, predstavlja regresioni problem. To znači da ćemo birati između modela koji su namenjeni za rešavanje regresionih problema poput Linearne Regresije, Random Forest, XGBoost, Support Vector Regression itd.

Za dobre perfomance linearne regresije potrebno je da budu zadovoljeni neki kriterijumi vezani za skup podataka nad kojima se vrši linearna regresija. Glavni zahtev je da bi trebalo da postoji linearna veza između prediktora X i odgovora Y. Ukoliko je promenljiva koja predstavlja odgovor Y *right-skewed*, tj. ukoliko se većina podataka nalazi sa leve strane histograma, onda je korisno uraditi logaritamsku transformaciju nad ciljanom promenljivom kako bi linearna regresija mogla da daje bolje rezultate.

```
ggplot(data.train, aes(x = SalePrice)) +
  geom_histogram(bins = 20, fill = "#1f78b4", color = "black", alpha = 0.7) +
  scale_x_continuous(labels = scales::comma) +
  labs(
    title = "Raspodela SalePrice",
    x = "Cena",
    y = "Broj kuća"
  ) +
  theme(plot.title = element_text(hjust = 0.5, face = "bold"))
```

Na slici ispod možemo videti da je promenljiva SalePrice right-skewed.



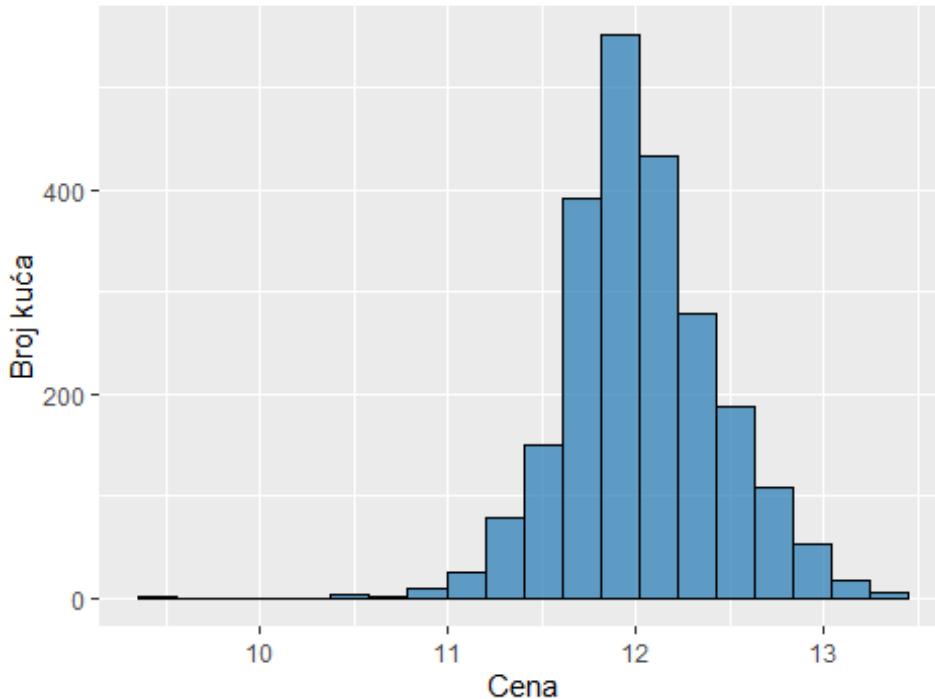
Uzimajući u obzir da ćemo da koristimo modele linearne regresije, primenićemo logaritamsku transformaciju nad kolonom SalePrice.

```
data.train = data.train %>% mutate(SalePrice = log1p(SalePrice))

ggplot(data.train, aes(x = SalePrice)) +
  geom_histogram(bins = 20, fill = "#1f78b4", color = "black", alpha = 0.7) +
  scale_x_continuous(labels = scales::comma) +
  labs(
    title = "Raspodela SalePrice",
    x = "Cena",
    y = "Broj kuća"
  ) +
  theme(plot.title = element_text(hjust = 0.5, face = "bold"))
```

Ispod možemo da vidimo histogram koji odgovara transformisanoj promenljivoj SalePrice, nakon primene logaritma.

### Raspodela SalePrice



Primećujemo da se na levom kraju histograma nalaze neke vrednosti koje odstupaju od ostatka vrednosti te ćemo ih pronaći i ukloniti kako bismo dobili histogram koji bi više odgovarao normalnoj raspodeli.

```
data.train %>% filter(SalePrice < 10) %>% select(0rder, SalePrice)

## # A tibble: 2 × 2
##   Order SalePrice
##   <int>     <dbl>
## 1 182      9.46
## 2 1554     9.48

data.train = data.train %>% filter(SalePrice >= 10)

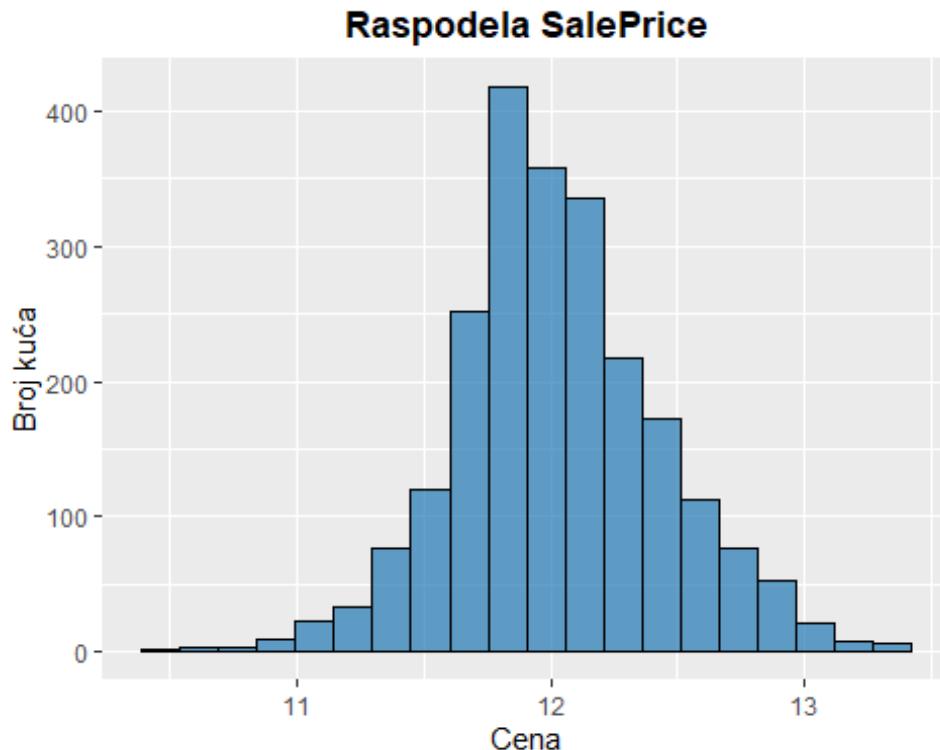
ggplot(data.train, aes(x = SalePrice)) +
  geom_histogram(bins = 20, fill = "#1f78b4", color = "black", alpha = 0.7) +
  scale_x_continuous(labels = scales::comma) +
  labs(
    title = "Raspodela SalePrice",
```

```

x = "Cena",
y = "Broj kuća"
) +
theme(plot.title = element_text(hjust = 0.5, face = "bold"))

```

Ispod možemo da vidimo histogram nakon uklonjenih vrednosti koje su odstupale.



Koristićemo iterativni proces kreiranja modela nazvan *forward-selection*. Taj proces podrazumeva inicijalno kreiranje modela sa samo jednim atributom, uglavnom onim koji najviše obećava, a zatim dodavanje jednog po jednog novog atributa u cilju smanjivanja greške modela. Naravno treba voditi računa da se izbegne overfitting, tj. preprilagođavanje modela podacima. Za odabir atributa u svakom koraku koristićemo znanje koje imamo o atributima iz odeljka EDA, kao i domensko znanje i dodatno ispitivanje korelacije između pojedinih atributa.

Za potrebe prvog modela odlučili smo se za atribut TotalSF zato što on poseduje najveći stepen korelacije sa atributom SalePrice od svih prediktora.

```

model1 = lm(SalePrice ~ TotalSF, data = data.train)
summary(model1)

##
## Call:
## lm(formula = SalePrice ~ TotalSF, data = data.train)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2.15835 -0.10701  0.02723  0.15471  0.85631
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 1.093e+01  1.657e-02   659.7 <2e-16 ***
## TotalSF     4.288e-04  6.242e-06    68.7  <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2312 on 2298 degrees of freedom
## Multiple R-squared:  0.6725, Adjusted R-squared:  0.6724
## F-statistic: 4720 on 1 and 2298 DF,  p-value: < 2.2e-16

```

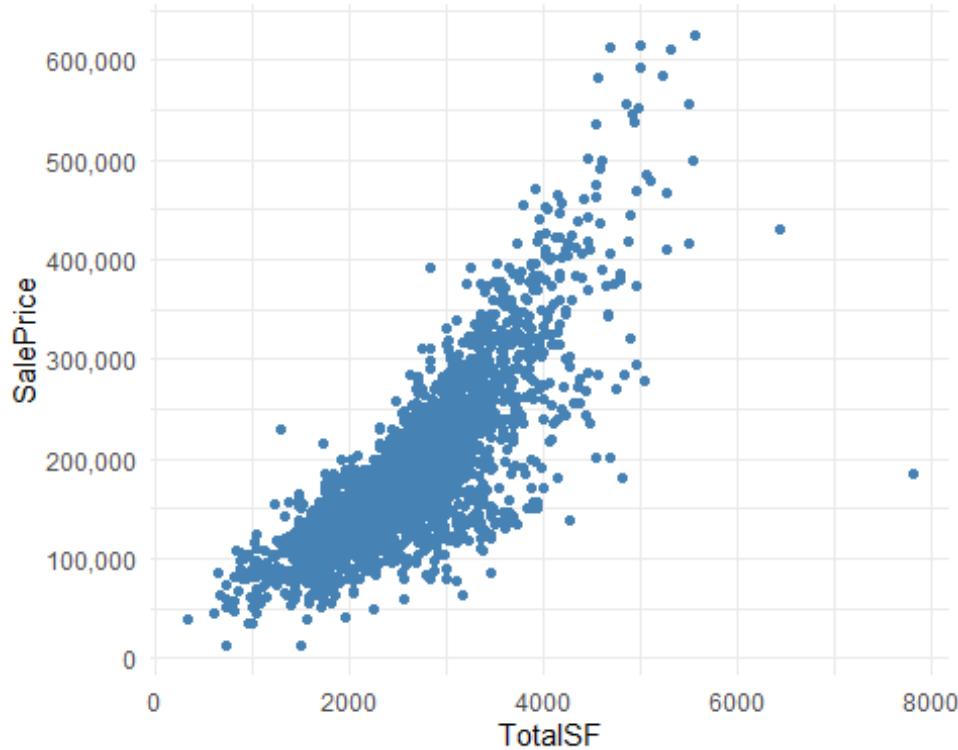
Možemo videti na da je atribut TotalSF i te kako značajan za predikciju promenljive SalePrice. Vrednost metrike  $R^2$  iznosi 0.6725 što znači da ovaj atribut sam objašnjava 67.25% varijabilnosti promenljive SalePrice. Metrika prilagođeni  $R^2$  je slična, samo što kažnjava dodavanje promenljivih koje ne doprinose mnogo predikcionoj moći modela.

Na grafiku ispod možemo i videti da zaista postoji linearna veza između promenljivih TotalSF i SalePrice, kao i da je koeficijent korelacije veliki.

```

ggplot(data = data[!is.na(data$SalePrice),], aes(x = TotalSF, y = SalePrice))
+
  geom_point(col = "steelblue") +
  scale_y_continuous(breaks = seq(0, 800000, by = 100000), labels = scales::comma) +
  theme_minimal()

```



Drugi atribut koji ćemo dodati je OverallQual zato što je on pokazao drugu najveću povezanost sa ciljanim atributom SalePrice.

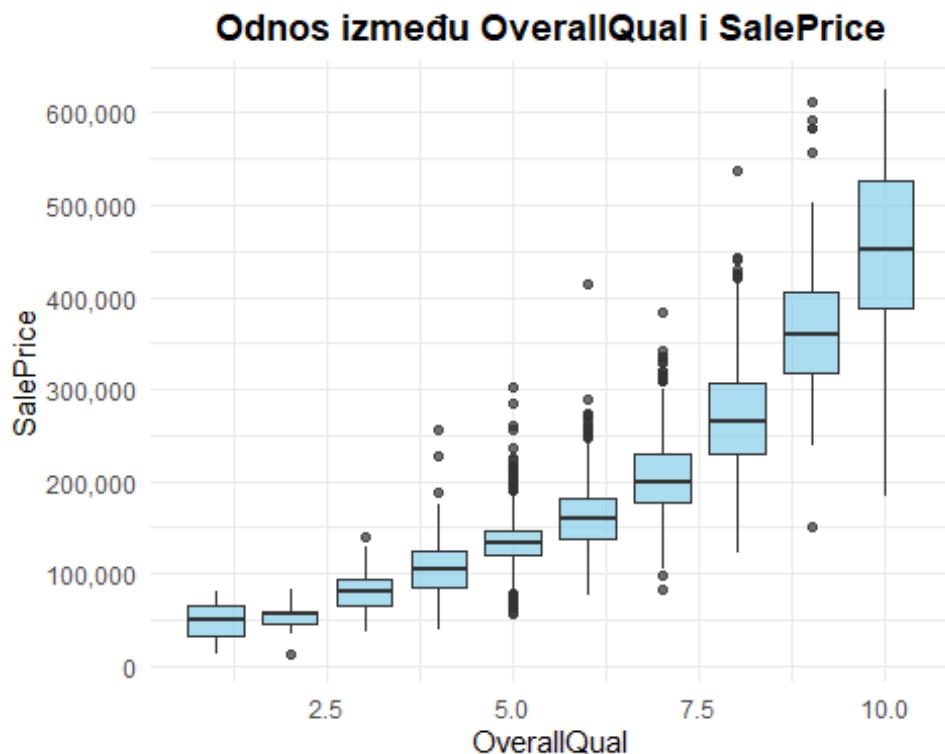
```
model2 = lm(SalePrice ~ TotalSF + OverallQual, data = data.train) summary(model2)

##
## Call:
## lm(formula = SalePrice ~ TotalSF + OverallQual, data = data.train)
##
## Residuals:
##      Min       1Q   Median       3Q      Max 
## -1.77391 -0.08446  0.01730  0.11093  0.61998 
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)    
## (Intercept) 1.050e+01  1.606e-02 653.89 <2e-16 ***
## TotalSF     2.481e-04  6.337e-06 39.15 <2e-16 ***
## OverallQual 1.459e-01  3.447e-03  42.32 <2e-16 ***
## ---
## Signif. codes:  0 '****' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1733 on 2297 degrees of freedom
```

```
## Multiple R-squared:  0.816, Adjusted R-squared:  0.8158
## F-statistic:  5093 on 2 and 2297 DF,  p-value: < 2.2e-16
```

Ovaj atribut takođe jako dobar za predikciju vrednosti SalePrice. Vrednost RSE se znatno smanjila što je poželjno. Vrednost  $R^2$  se dosta povećala i sada naš model sa samo ova 2 atributa objašnjava 81.6% varijabilnosti promenljive SalePrice.

```
ggplot(data, aes(x = OverallQual, y = SalePrice, group = OverallQual)) +
  geom_boxplot(fill = "skyblue", alpha = 0.7) +
  scale_y_continuous(breaks = seq(0, 800000, by = 100000), labels = scales::comma) +
  labs(
    title = "Odnos između OverallQual i SalePrice",
    x = "OverallQual",
    y = "SalePrice"
  ) +
  theme_minimal() +
  theme(plot.title = element_text(hjust = 0.5, face = "bold"))
```



Na grafiku iznad možemo da vidimo da i te kako postoji jaka korelacija između promenljive OverallQual i SalePrice.

Sada tražimo nov atribut koji ćemo dodati u naš model. Uzimamo u obzir da multikolinearnost između prediktora loše utiče na model višestruke linearne regresije te ćemo se odlučiti za atribut koji nije u velikoj korelaciji sa već dodatim atributima TotalSF i OverallQual.

```
cor(data.train$TotalSF, data.train$TotalFinishedSF)
## [1] 0.8273132
cor(data.train$OverallQual, data.train$TotalFinishedSF)
## [1] 0.5480203
cor(data.train$TotalSF, data.train$GrLivArea)
## [1] 0.8671521
cor(data.train$TotalSF, data.train$GarageCars)
## [1] 0.5750082
cor(data.train$OverallQual, data.train$GarageCars)
## [1] 0.6075288
```

Sledeći atribut za koji smo se odlučili je GarageCars. Njega dodajemo u sledeći model. S obzirom na to da ima poprilično modela nećemo za svaki da prikazujemo summary, već tek nakon svakih nekoliko dodatih atributa da bismo ispratili kako naš model napreduje.

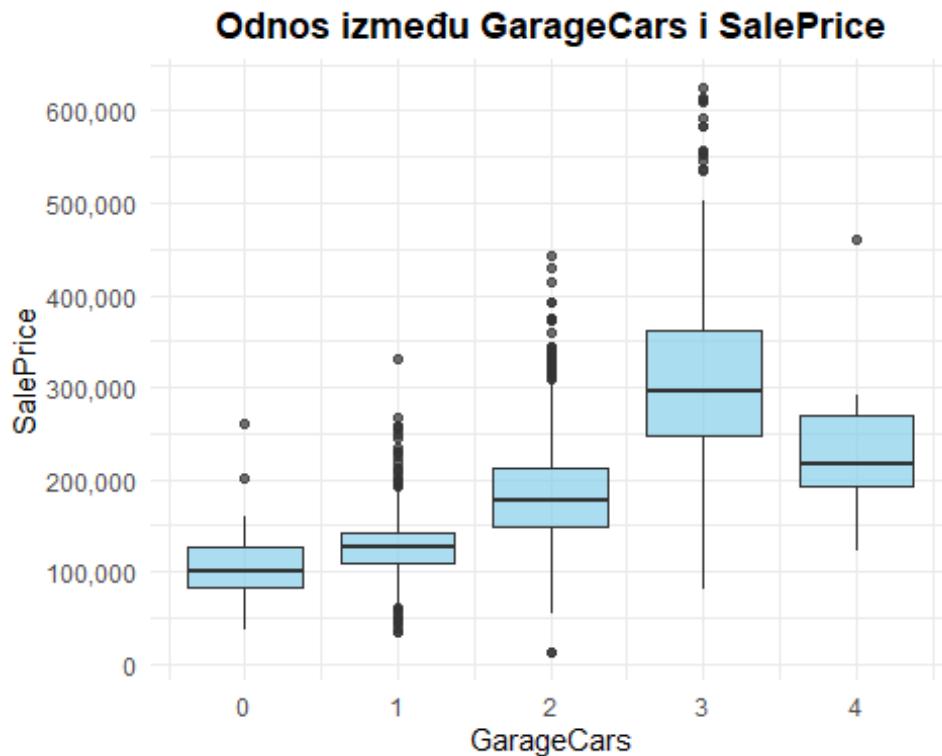
```
model3 = lm(SalePrice ~ TotalSF + OverallQual + GarageCars, data = data.train)
```

```
ggplot(data, aes(x = GarageCars, y = SalePrice, group = GarageCars)) +
  geom_boxplot(fill = "skyblue", alpha = 0.7) +
  scale_y_continuous(breaks = seq(0, 800000, by = 100000), labels = scales::comma) +
  labs(
    title = "Odnos između GarageCars i SalePrice",
    x = "GarageCars",
```

```

y = "SalePrice"
) +
theme_minimal() +
theme(plot.title = element_text(hjust = 0.5, face = "bold"))

```



Sa grafika možemo da vidimo da i promenljiva GarageCars ima veliki uticaj na promenljivu SalePrice i da je dobro uvrstiti je u prediktore modela.

Za naš sledeći model smo se odlučili da dodamo promenljivu TotalBaths, na osnovu koje takođe može solidno da se proceni vrednost promenljive SalePrice.

```

model4 = lm(SalePrice ~ TotalSF + OverallQual + GarageCars + TotalBaths, data
= data.train)

```

U naredni model smo dodali promenljivu HouseAge, za koju je logično da će imati uticaj na promenljivu SalePrice. Summary za ovaj model je prikazan i možemo videti kakva je predikciona moć našeg novog modela.

```
model5 = lm(SalePrice ~ TotalSF + OverallQual + GarageCars + TotalBaths + HouseAge, data = data.train)
summary(model5)

##
## Call:
## lm(formula = SalePrice ~ TotalSF + OverallQual + GarageCars +
##     TotalBaths + HouseAge, data = data.train)
##
## Residuals:
##      Min        1Q    Median        3Q       Max
## -1.62590 -0.07569  0.00928  0.08802  0.56053
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 1.071e+01  2.235e-02 479.205 <2e-16 ***
## TotalSF     2.115e-04  6.137e-06 34.454 <2e-16 ***
## OverallQual 1.008e-01  3.540e-03 28.481 <2e-16 ***
## GarageCars   6.917e-02  5.829e-03 11.867 <2e-16 ***
## TotalBaths   5.855e-02  6.890e-03  8.497 <2e-16 ***
## HouseAge    -1.909e-03  1.432e-04 -13.332 <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1525 on 2294 degrees of freedom
## Multiple R-squared:  0.8578, Adjusted R-squared:  0.8575
## F-statistic: 2768 on 5 and 2294 DF, p-value: < 2.2e-16

vif(model5)

##      TotalSF OverallQual GarageCars TotalBaths      HouseAge
## 2.222761  2.498585  1.910912  1.624804  1.856123
```

Možemo videti da su sve promenljive koje smo u međuvremenu dodali odlični prediktori za ciljanu promenljivu SalePrice. Vidimo da je vrednost metrike  $R^2$  sada 0.8578 i da je RSE spala na 0.1525.

Takođe treba obratiti pažnju na potencijalnu multikolinearnost između prediktora u višestrukoj linearnoj regresiji. Možemo iskoristiti funkciju `vif()` da proverimo multikolinearnost. Ukoliko je vrednost koeficijenta  $> 5$  smatra se da ne postoji značajna multikolinearnost među prediktorima modela, što je ovde slučaj.

```
model6 = lm(SalePrice ~ TotalSF + OverallQual + GarageCars + TotalBaths + HouseAge + Neighborhood, data = data.train)
```

```
model7 = lm(SalePrice ~ TotalSF + OverallQual + GarageCars + TotalBaths + HouseAge + Neighborhood + TotRmsAbvGrd, data = data.train)
```

U naredna dva modela smo dodali promenljive Neighborhood i TotRmsAbvGrd. Proverićemo koeficijent korelacije atributa TotRmsAbvGrd sa ciljanom promenljivom SalePrice.

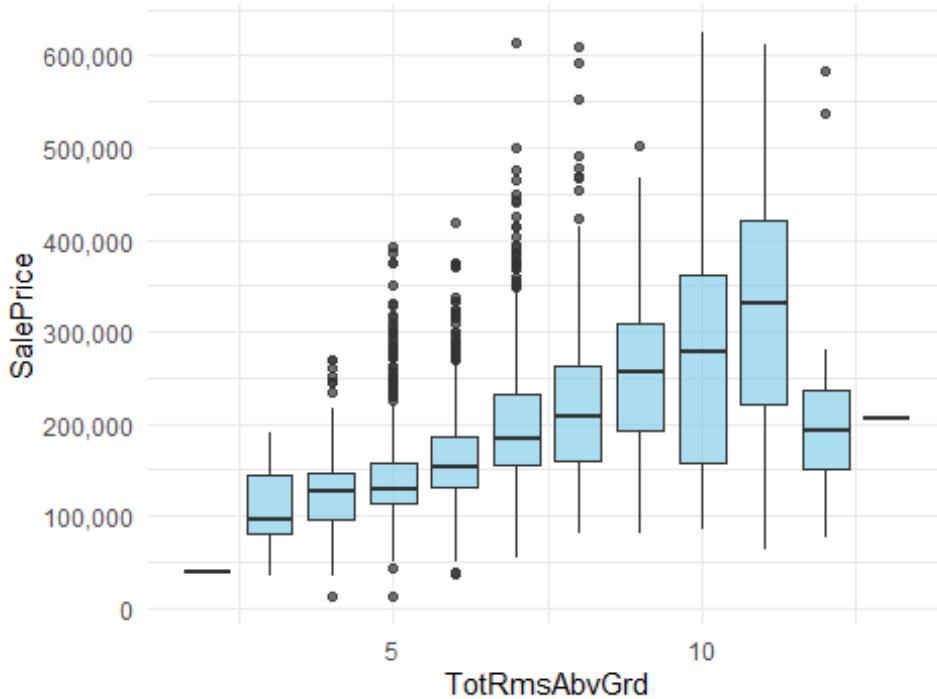
```
cor(data.train$TotalSF, data.train$TotRmsAbvGrd)

## [1] 0.6569158

ggplot(data, aes(x = TotRmsAbvGrd, y = SalePrice, group = TotRmsAbvGrd)) +
  geom_boxplot(fill = "skyblue", alpha = 0.7) +
  scale_y_continuous(breaks = seq(0, 800000, by = 100000), labels = scales::comma) +
  labs(
    title = "Odnos između TotRmsAbvGrd i SalePrice",
    x = "TotRmsAbvGrd",
    y = "SalePrice"
  ) +
  theme_minimal() +
  theme(plot.title = element_text(hjust = 0.5, face = "bold"))
```

Vidimo da koeficijent korelacije iznosi približno 0.657 što sugeriše da i ovaj prediktor može da bude dobar izbor za naš model. Ispod je data slika grafika odnosa ove dve promenljive.

## Odnos između TotRmsAbvGrd i SalePrice



Naredni model je uzeo u obzir i potencijalnu sinergiju između promenljivih TotalSF i TotRmsAbvGrd bez dodavanja novih promenljivih.

```
model8 = lm(SalePrice ~ TotalSF * TotRmsAbvGrd + OverallQual + GarageCars + TotalBaths + HouseAge + Neighborhood, data = data.train)
summary(model8)

##
## Call:
## lm(formula = SalePrice ~ TotalSF * TotRmsAbvGrd + OverallQual +
##     GarageCars + TotalBaths + HouseAge + Neighborhood, data = data.train)
##
## Residuals:
##      Min        1Q    Median        3Q       Max 
## -1.13514 -0.06523  0.01013  0.07871  0.56924 
##
## Coefficients:
##                               Estimate Std. Error t value Pr(>|t|)    
## (Intercept)             1.047e+01  5.250e-02 199.415 < 2e-16 ***
## TotalSF                  2.873e-04  1.603e-05 17.925 < 2e-16 ***
## TotRmsAbvGrd              4.364e-02  6.108e-03  7.144 1.21e-12 ***
## OverallQual                9.112e-02  3.714e-03 24.530 < 2e-16 ***
## GarageCars                 6.218e-02  5.584e-03 11.134 < 2e-16 ***
## TotalBaths                 5.794e-02  6.553e-03  8.842 < 2e-16 ***
```

```

## HouseAge      -1.615e-03  2.193e-04  -7.364  2.49e-13 ***
## NeighborhoodBlueste -3.614e-03  6.466e-02  -0.056  0.955432
## NeighborhoodBrDale -1.134e-01  4.179e-02  -2.713  0.006714 **
## NeighborhoodBrkSide 7.254e-02  3.621e-02   2.003  0.045285 *
## NeighborhoodClearCr 1.833e-01  3.862e-02   4.746  2.20e-06 ***
## NeighborhoodCollgCr 8.249e-02  3.038e-02   2.715  0.006677 **
## NeighborhoodCrawfor 2.208e-01  3.509e-02   6.291  3.77e-10 ***
## NeighborhoodEdwards 3.302e-02  3.311e-02   0.998  0.318620
## NeighborhoodGilbert 7.169e-02  3.150e-02   2.276  0.022949 *
## NeighborhoodGreens 7.052e-02  6.137e-02   1.149  0.250631
## NeighborhoodGrnHill 5.829e-01  1.039e-01   5.609  2.28e-08 ***
## NeighborhoodIDOTRR -5.824e-02  3.743e-02  -1.556  0.119883
## NeighborhoodMeadowV -6.584e-02  4.034e-02  -1.632  0.102782
## NeighborhoodMitchel 8.286e-02  3.317e-02   2.498  0.012572 *
## NeighborhoodNAmes 7.376e-02  3.145e-02   2.345  0.019100 *
## NeighborhoodNoRidge 1.650e-01  3.574e-02   4.616  4.12e-06 ***
## NeighborhoodNPkVill -3.908e-02  4.382e-02  -0.892  0.372625
## NeighborhoodNridgHt 1.795e-01  3.167e-02   5.669  1.62e-08 ***
## NeighborhoodNWAmes 4.090e-02  3.269e-02   1.251  0.211064
## NeighborhoodOldTown 7.212e-03  3.525e-02   0.205  0.837916
## NeighborhoodSawyer 7.920e-02  3.292e-02   2.406  0.016197 *
## NeighborhoodSawyerW 4.977e-02  3.243e-02   1.535  0.125007
## NeighborhoodSomerst 8.721e-02  3.106e-02   2.808  0.005032 **
## NeighborhoodStoneBr 2.251e-01  3.709e-02   6.071  1.49e-09 ***
## NeighborhoodSWISU 3.015e-02  3.973e-02   0.759  0.448000
## NeighborhoodTimber 1.184e-01  3.447e-02   3.436  0.000601 ***
## NeighborhoodVeenker 1.243e-01  4.686e-02   2.652  0.008062 **
## TotalSF:TotRmsAbvGrd -1.440e-05  2.033e-06  -7.082  1.89e-12 ***
## ---
## Signif. codes:  0 '****' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1408 on 2266 degrees of freedom
## Multiple R-squared:  0.8802, Adjusted R-squared:  0.8785
## F-statistic: 504.6 on 33 and 2266 DF,  p-value: < 2.2e-16

```

U naredna dva modela smo dodali atribute KitchenQual i Fireplaces.

```

model9 = lm(SalePrice ~ TotalSF + OverallQual + GarageCars + TotalBaths + HouseAge + Neighborhood +
            KitchenQual, data = data.train)

```

```

model10 = lm(SalePrice ~ TotalSF + OverallQual + GarageCars + TotalBaths + HouseAge + Neighborhood +
            KitchenQual + Fireplaces, data = data.train)

```

U finalni model smo dodali promenljivu HasGarage i uzeli u obzir potencijalnu sinergiju između promenljivih TotalSF i OverallQual.

```

model11 = lm(SalePrice ~ TotalSF * OverallQual + GarageCars + TotalBaths + HouseAge + HasGarage +
             Neighborhood + KitchenQual + Fireplaces, data = data.train)
summary(model11)

##
## Call:
## lm(formula = SalePrice ~ TotalSF * OverallQual + GarageCars +
##     TotalBaths + HouseAge + HasGarage + Neighborhood + KitchenQual +
##     Fireplaces, data = data.train)
##
## Residuals:
##      Min        1Q    Median        3Q       Max
## -1.22127 -0.06612  0.00977  0.08032  0.48632
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)    
## (Intercept) 1.044e+01 5.459e-02 191.292 < 2e-16 ***
## TotalSF      2.305e-04 1.569e-05 14.685 < 2e-16 ***
## OverallQual 9.802e-02 6.722e-03 14.582 < 2e-16 ***
## GarageCars   4.438e-02 6.494e-03  6.833 1.06e-11 ***
## TotalBaths   5.085e-02 6.292e-03  8.082 1.03e-15 ***
## HouseAge    -1.463e-03 2.103e-04 -6.958 4.50e-12 ***
## HasGarage1   5.904e-02 1.645e-02  3.589 0.000340 ***
## NeighborhoodBlueste -3.399e-03 6.234e-02 -0.055 0.956522
## NeighborhoodBrDale -8.920e-02 4.047e-02 -2.204 0.027621 *
## NeighborhoodBrkSide 8.393e-02 3.501e-02  2.397 0.016596 *
## NeighborhoodClearCr 1.735e-01 3.733e-02  4.649 3.53e-06 ***
## NeighborhoodCollgCr 1.029e-01 2.939e-02  3.500 0.000475 ***
## NeighborhoodCrawfor 2.118e-01 3.395e-02  6.238 5.27e-10 ***
## NeighborhoodEdwards 5.194e-02 3.202e-02  1.622 0.104962
## NeighborhoodGilbert 9.386e-02 3.035e-02  3.092 0.002011 ** 
## NeighborhoodGreens  1.379e-02 5.886e-02  0.234 0.814749
## NeighborhoodGrnHill 5.492e-01 1.002e-01  5.479 4.76e-08 ***
## NeighborhoodIDOTRR -3.447e-02 3.626e-02 -0.951 0.341890
## NeighborhoodMeadowV -6.562e-02 3.891e-02 -1.686 0.091861 .
## NeighborhoodMitchel 1.165e-01 3.210e-02  3.629 0.000290 ***
## NeighborhoodNAmes   8.700e-02 3.046e-02  2.856 0.004331 **
## NeighborhoodNoRidge  1.876e-01 3.470e-02  5.406 7.12e-08 ***
## NeighborhoodNPkVill -1.463e-02 4.238e-02 -0.345 0.729994
## NeighborhoodNridgHt  1.811e-01 3.092e-02  5.856 5.42e-09 ***
## NeighborhoodNWAmes   6.500e-02 3.164e-02  2.054 0.040071 *
## NeighborhoodOldTown  2.251e-02 3.409e-02  0.660 0.509093
## NeighborhoodSawyer   9.427e-02 3.186e-02  2.959 0.003123 **
## NeighborhoodSawyerW  6.862e-02 3.130e-02  2.192 0.028467 *

```

```

## NeighborhoodSomerst 1.074e-01 3.005e-02 3.576 0.000357 ***
## NeighborhoodStoneBr 2.373e-01 3.609e-02 6.577 5.93e-11 ***
## NeighborhoodSWISU 5.084e-02 3.836e-02 1.325 0.185205
## NeighborhoodTimber 1.386e-01 3.329e-02 4.164 3.24e-05 ***
## NeighborhoodVeenker 1.273e-01 4.520e-02 2.816 0.004911 **
## KitchenQual 6.757e-02 6.275e-03 10.768 < 2e-16 ***
## Fireplaces 5.208e-02 5.418e-03 9.614 < 2e-16 ***
## TotalSF:OverallQual -8.481e-06 2.290e-06 -3.704 0.000218 ***
## ---
## Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1357 on 2264 degrees of freedom
## Multiple R-squared: 0.8888, Adjusted R-squared: 0.8871
## F-statistic: 517.1 on 35 and 2264 DF, p-value: < 2.2e-16

vif(model11)

## there are higher-order terms (interactions) in this model
## consider setting type = 'predictor'; see ?vif

##          GVIF Df GVIF^(1/(2*Df))
## TotalSF 18.342489 1    4.282813
## OverallQual 11.368506 1    3.371722
## GarageCars 2.993578 1    1.730196
## TotalBaths 1.709852 1    1.307613
## HouseAge  5.053517 1    2.248003
## HasGarage  1.671766 1    1.292968
## Neighborhood 14.541110 26   1.052829
## KitchenQual 2.219737 1    1.489878
## Fireplaces  1.490915 1    1.221030
## TotalSF:OverallQual 43.389776 1    6.587092

```

Možemo da primetimo da su svi atributi koje smo dodali pokazali veliku značajnost za predikciju vrednosti ciljane promenljive SalePrice. Konačna vrednost metrike  $R^2$  iznosi 0.8888 što nam govori da naš model objašnjava 88.88% varijabilnosti promenljive SalePrice korišćenjem navedenih prediktora. Konačna vrednost greške RSE iznosi 0.1357 što je takođe jako dobro. Vidimo da vif koeficijenti pokazuju da nema značajne multikolinearnosti između prediktora.

Sledeće prelazimo na regularizovane modele linearne regresije zvane Ridge i Lasso. Ovi modeli modifikuju jednačinu osnovne linearne regresije kako bi izbegli overfitting i popravili moć generalizacije. Uključuju kazneni parametar lambda koji ima ulogu da smanji vrednosti određenih težinskih koeficijenata koji idu uz odgovarajuće prediktore. Ridge nam daje pojedine koeficijente približno jednakе nuli, kako bi drastično smanjio uticaj nekih atributa u jednačini, dok ih Lasso izjednačava sa nulom kako bi ih skroz uklonio iz jednačine.

```

x_train <- model.matrix(SalePrice ~ ., data = data.train)[, -1]
y_train <- data.train$SalePrice

for (col in names(data.test)) {
  if (is.factor(data.train[[col]])) {
    data.test[[col]] <- factor(data.test[[col]], levels = levels(data.train[[col]]))
  }
}

x_test <- model.matrix(~ ., data = data.test)[, -1]
x_test <- x_test[, colnames(x_train)]
y_test <- data.test$SalePrice

```

Koristimo metodu unakrsne validacije (cross-validation) da bismo što bolju vrednost za naš parametar lambda koji minimizuje grešku modela. Možemo videti da je Ridge modelu odgovaraju vrednosti metrika RMSE = 18774.8 i MAE = 13329.79, dok je Lasso model imao malo bolje rezultate sa vrednostima RMSE = 18485.85 i MAE = 13109.65.

```

set.seed(123)
ridge_cv <- cv.glmnet(x_train, y_train, alpha = 0)
lasso_cv <- cv.glmnet(x_train, y_train, alpha = 1)
ridge_cv$lambda.min

## [1] 0.07766592

lasso_cv$lambda.min

## [1] 0.002427616

ridge_preds = expm1(predict(ridge_cv, newx = x_test, s = "lambda.min"))
lasso_preds = expm1(predict(lasso_cv, newx = x_test, s = "lambda.min"))

rmse(y_test, ridge_preds)

## [1] 18774.8

mae(y_test, ridge_preds)

## [1] 13329.79

rmse(y_test, lasso_preds)

## [1] 18485.85

mae(y_test, lasso_preds)

## [1] 13109.65

```

Sledeći na redu je model Random Forest. Ova metoda se zasniva na izgradnji više Decision Tree modela i kombinovanju njihovih predikcija kako bi se došlo do što bolje predikcije. Na ovaj način se izbegava overfitting i poboljšava se moć generalizacije modela. Ova vrsta modela bolje hvata nelinearne odnose između podataka nego modeli linearne regresije. Vidimo da je ovaj model imao lošije performanse nego dosadašnji sa vrednostima metrika RMSE = 25591.85 i MAE = 16300.5. Model bi dao bolje rezultate da je korišćen veći broj stabala za odlučivanje ali to negativno utiče na brzinu formiranja modela, stoga je korišćeno samo 50 stabala u ovom slučaju.

```
library(randomForest)

## randomForest 4.7-1.2

## Type rfNews() to see new features/changes/bug fixes.

##
## Attaching package: 'randomForest'

## The following object is masked from 'package:dplyr':
## 
##     combine

## The following object is masked from 'package:ggplot2':
## 
##     margin

dummies <- dummyVars(~ ., data = data.train[, !names(data.train) %in% "SalePrice"])

data.train.num <- data.frame(predict(dummies, newdata = data.train))
data.train.num$SalePrice <- data.train$SalePrice

data.test.num <- data.frame(predict(dummies, newdata = data.test))

rf_model <- randomForest(SalePrice ~ ., data = data.train.num, ntree = 50, mtry = 10, importance = TRUE)

rf_preds <- expm1(predict(rf_model, newdata = data.test.num))

rmse(data.test$SalePrice, rf_preds)

## [1] 25591.85

mae(data.test$SalePrice, rf_preds)

## [1] 16300.5
```

Sledeći model na redu je XGBoost. On se takođe zasniva na stablima odlučivanja kao i njegov prethodnik Random Forest, samo što se ovde stabla ne treniraju nezavisno i onda uzima njihov prosek, već se stabla treniraju sekvencijalno i svako sledeće stablo gleda da ispravi grešku koju pravi prethodno. Ovaj model mašinskog učenja se odlično pokazao na velikim skupovima tabelarnih podataka (poput onih na Kaggle-u). Rezultuje velikom preciznošću i može da uhvati i nelinarne odnose među podacima. Takođe poseduje ugrađene metode za regularizaciju koji sprečavaju overfitting.

```
library(xgboost)

train_idx <- sample(seq_len(nrow(data)), size = 0.8 * nrow(data))
train_data <- data[train_idx, ]
val_data   <- data[-train_idx, ]

dummies <- dummyVars(~ ., data = train_data[, !names(train_data) %in% "SalePrice"])
train_data_num <- data.frame(predict(dummies, newdata = train_data))
train_data_num$SalePrice <- train_data$SalePrice

train_data_num$SalePrice <- log1p(train_data_num$SalePrice)

val_data_num <- data.frame(predict(dummies, newdata = val_data))

train_matrix <- as.matrix(train_data_num[, !names(train_data_num) %in% "SalePrice"])
train_label   <- train_data_num$SalePrice

val_matrix <- as.matrix(val_data_num)
val_label   <- val_data$SalePrice

xgb_model = xgboost(data = train_matrix, label = train_label, nrounds = 100,
objective = "reg:squarederror", verbose = 0)

pred_xgb = expm1(predict(xgb_model, newdata = val_matrix))

rmse(val_data$SalePrice, pred_xgb)
## [1] 24570.33

mae(val_data$SalePrice, pred_xgb)
## [1] 13785.05
```

Vrednosti metrika za model XGBoost iznose RMSE = 24570.33 i MAE = 13785.05 što je u rangu sa njegovim prethodnikom Random Forest.

Poslednji model koji ćemo posmatrati je Support Vector Regression (SVR). Ovaj algoritam se zasniva na Support Vector Machine (SVM) koji je njegov pandan za klasifikacione probleme. SVM pronalazi hiperravan koja se najbolje prilagođava podacima uz određenu granicu greške. Za razliku od linearne regresije, koja pokušava da minimizuje sumu razlika kvadrata, ovaj algoritam pokušava da pronađe pravu sa određenom širinom gde se male greške ignorisu, a veće greške kažnjavaju. Ovakav model je otporniji na mali šum u podacima u odnosu na ostale modele koje smo uzeli u obzir. Ovaj model jako loše radi sa nenormalizovanim podacima, pa smo iste skalirali i zatim trenirali i evaluirali model. On se pokazao najlošijim među svim regresionim modelima. Vrednost metrika iznosi RMSE = 52333.98 i MAE = 30229.96.

```
library(e1071)

##
## Attaching package: 'e1071'

## The following object is masked from 'package:ggplot2':
##
##     element

train_data = data.train[sapply(data.train, is.numeric)]
test_data = data.test[sapply(data.test, is.numeric)]

X_train = train_data %>% select(-SalePrice)
y_train = train_data$SalePrice

X_test = test_data %>% select(-SalePrice)
y_test = test_data$SalePrice

scaler_X <- preprocess(X_train, method = c("center", "scale"))
X_train_scaled <- predict(scaler_X, X_train)
X_test_scaled <- predict(scaler_X, X_test)

y_train_scaled <- scale(y_train)

y_mean <- attr(y_train_scaled, "scaled:center")
y_sd   <- attr(y_train_scaled, "scaled:scale")

svr_model <- svm(
  x = X_train_scaled,
  y = y_train_scaled,
  type = "eps-regression",
  kernel = "radial",
  cost = 10,
  gamma = 0.1
)
```

```
pred_scaled <- predict(svr_model, newdata = X_test_scaled)
pred_original <- pred_scaled * y_sd + y_mean
pred_svr <- expm1(pred_original)

rmse(data.test$SalePrice, pred_svr)
## [1] 52333.98

mae(data.test$SalePrice, pred_svr)
## [1] 30229.96
```

## Poređenje modela

U ovom odeljku ćemo uporediti sve modele koje smo koristili za rešavanje problema predviđanja cena kuće (SalePrice) u zavisnosti od ostalih atributa. U tabeli ispod su u prvoj koloni su navedeni nazivi modela dok su u drugoj i trećoj navedene metrike RMSE i MAE za odgovarajuće modele, respektivno.

Model	RMSE	MAE
Ridge Regression	18,774.8	13,329.79
Lasso Regression	18,485.85	13,109.65
Random Forest	25,591.85	16,300.5
XGBoost	24,570.33	13,785.05
Support Vector Regressor	52,333.98	30,229.96

Možemo da primetimo da je najmanju RMSE, kao i najmanju MAE imao model Lasso Regression koji se najbolje pokazao za potrebe rešavanja ovog problema predviđanja cene kuća. Ne mnogo lošiji od njega bio je model Ridge Regression sa malo većim vrednostima obe metrike. Nakon njih slične rezultate su dali modeli Random Forest i XGBoost sa malim međusobnim varijacijama u vrednostima metrika. Najlošiji model je u ovom slučaju bio Support Vector Regressor koji je imao daleko najveće vrednosti za RMSE, kao i za MAE.

Svaki od ovih modela ima svoje prednosti i mane i to što su se neki pokazali bolje a neki lošije može da bude uzrok više faktora poput linearnosti ili nelinearnosti odnosa između podataka, otpornosti na šum u podacima, nedovoljna komputaciona moć u vidu manjeg broja stabla za treniranje modela, itd.

## Zaključak

U ovom radu smo se bavili predviđanjem cena kuća korišćenjem različitih modela. Proces je obuhvatio prikupljanje, čišćenje i pripremu podataka, eksplorativnu analizu (EDA), feature engineering, izgradnju i evaluaciju modela.

Analiza podataka je pokazala da su najznačajniji faktori koji utiču na cenu kuće ukupna površina (TotalSF), kvalitet kuće (OverallQual), broj garažnih mesta, površina prostora iznad zemlje, starost kuće, kvalitet kuhinje, kao i lokacija (Neighborhood). Kreiranjem novih promenljivih, poput TotalFinishedSF, HouseAge, HouseRemodAge, binarnih promenljivih za prisustvo podruma, garaže, kamina i bazena, unapredili smo prediktivnu moć modela.

Primena neregularizovane višestruke linearne regresije sa transformisanom ciljnom promenljivom ( $\log(\text{SalePrice})$ ) pokazala je da jednostavan model sa nekoliko pažljivo odabralih promenljivih može objasniti preko 88% varijabilnosti cena.

Rezultati prikazuju da su Ridge i Lasso regresija ostvarile najbolje performanse u predikciji cena kuća, sa najmanjim vrednostima RMSE i MAE metrika. Ovi modeli su se pokazali stabilnim i pouzdanim za predviđanje. Random Forest i XGBoost su postigli nešto veće greške, ali su i dalje pokazali solidne rezultate i mogu se dodatno poboljšati podešavanjem parametara. Support Vector Regressor je pokazao najlošije rezultate među izabranim modelima. Na osnovu dobijenih rezultata može se zaključiti da pravilna priprema podataka i izbor odgovarajućeg modela imaju presudnu ulogu u tačnosti predikcije cena kuća.

Dodatno, projekat bi se mogao unaprediti kroz nekoliko koraka. U fazi feature engineeringa mogla bi se kreirati nova promenljiva, poput TotalQual, koja bi objedinjavala ukupni kvalitet kuće na osnovu više postojećih atributa. Takođe, za potrebe modela Support Vector Regression (SVR) mogli bi da budu uključeni i kategoriski atributi što bi dodatno doprinelo njegovoj predikcionoj moći. Pored toga, za preciznije otkrivanje outliera mogao bi se koristiti Z-score pristup, čime bi se smanjio subjektivni uticaj vizuelne procene.

Ovaj rad pokazuje da pravilna analiza i priprema podataka, kao i kvalitetan feature engineering, zajedno sa eksplorativnom analizom podataka imaju ključnu ulogu u kreiranju preciznih modela za predikciju cena kuća. Linearna regresija, uz dobar odabir promenljivih i transformaciju ciljne promenljive, pokazala se kao efikasan i interpretabilan pristup za ovaj problem, dok napredniji modeli mogu biti korisni za dodatno poboljšanje performansi.

## Literatura

- [1] Wickham, H., & Grolemund, G. (2017). R for Data Science: Import, Tidy, Transform, Visualize, and Model Data. O'Reilly Media.
- [2] Microsoft Teams, kanal Uvod u nauku o podacima
- [3] <https://www.kaggle.com/competitions/house-prices-advanced-regression-techniques/>
- [4] James, G., Witten, D., Hastie, T., Tibshirani, R. (2021). An Introduction to Statistical Learning: with Applications in R (2nd edition).
- [5] Tidy Modeling with R: A Framework for Modeling in the Tidyverse, 1st Edition, Max Kuhn, Julia Silge.