I-SUNS: Zadanie č. 2

Stromy, stroje, hlasovania a redukcia dimenzie

Matúš Vetrík

Úlohou práce je zanalyzovať dáta o domoch a vytvoriť sieť, ktorá vie čo najpresnejšie uhádnuť cenu domu podľa jeho atribútov.

Na riešenie tohto zadania bol zvolený programovací jazyk Python a viacero pomocných knižníc. Na Na úpravu a spracovanie dát boli použité knižnice ***Pandas*** a ***Numpy***. Na graficke zobrazenie dát boli použité knižnice ***Plotly*****,** ***Matplotlib*****,** ***Yellowbrick*** a vizualizačné funkcie od ***Sklearn*****.** Na trénovanie, vytváranie modelov a reduckcie dimenzií bola použitá knižnica ***Sklearn***. Projekt je vytváraný v prostredí ***Jupyter***.

Spracovanie dát

Pôvodné dáta train.csv a test.csv nie sú v ideálnej forme na nasledujúce cvičenia. Mnoho stĺpcov obsahuje hodnoty v textovej forme a v stĺpcoch sa vyskutujú nekonzistentné číselne hodnoty. Prvý problém riešia dva súbory train_dummy.csv a test_dummy.csv. Tieto súbory obsahujú rovnaké dáta ako pôvodné súbory ale všetky stĺpce, ktoré obsahovali textové hodnoty sú one hot encodnuté, čiže každá textová hodnota má vlastný stĺpec s true alebo false hodnotou. Z pôvodných 73 stĺpcov vzniklo 256 vďaka one hot encodingu. Druhý problém sme vyriešili škálovaním dát. Na škálovanie sme použili funkciu **StandardScaler** poskytovanú knižnicou ***sklearn.preprocessing***. Škálovaním dosiahneme, že všetký hodnoty v stĺpcoch sú upravené na menšie čísla a to spôsobom, že nová hodnota sa vypočíta na základe originálnej hodnoty, mediánu hodnôt a smerodajnej odchýlky. Na nasledujúcich ukážkach môžme vidieť ako dáta vyzerali pred a po škálovaní.

	MSSubClass	LotFrontage	LotArea	OverallCond	YearBuilt	YearRemodAdd	\
0	90	87.0	9246	5	1973	1973	
1	60	104.0	21535	6	1994	1995	
2	60	86.0	10380	5	1986	1987	
3	50	52.0	6240	6	1934	1950	
4	20	74.0	8532	6	1954	1990	

	MasVnrArea	BsmtFinSF1	BsmtFinSF2	BsmtUnfSF	...	SaleType_ConLw	\
0	564.0	0	0	1656	...	0	
1	1170.0	1455	0	989	...	0	
2	172.0	28	1474	0	...	0	
3	0.0	0	0	816	...	0	
4	650.0	1213	0	84	...	0	

	MSSubClass	LotFrontage	LotArea	OverallCond	YearBuilt	YearRemodAdd	\
0	0.838946	0.666742	-0.115746	-0.541653	0.027184	-0.604075	
1	0.117217	1.362390	1.290914	0.392575	0.702176	0.447375	
2	0.117217	0.625822	0.014058	-0.541653	0.445037	0.065029	
3	-0.123359	-0.765474	-0.459827	0.392575	-1.226372	-1.703319	
4	-0.845088	0.134776	-0.197474	0.392575	-0.583523	0.208409	

	MasVnrArea	BsmtFinSF1	BsmtFinSF2	BsmtUnfSF	...	SaleType_ConLw	\
0	2.332453	-0.955347	-0.295898	2.387890	...	-0.065372	
1	5.453516	2.123390	-0.295898	0.876841	...	-0.065372	
2	0.313547	-0.896100	8.600367	-1.363679	...	-0.065372	
3	-0.572299	-0.955347	-0.295898	0.484920	...	-0.065372	
4	2.775376	1.611325	-0.295898	-1.173382	...	-0.065372	

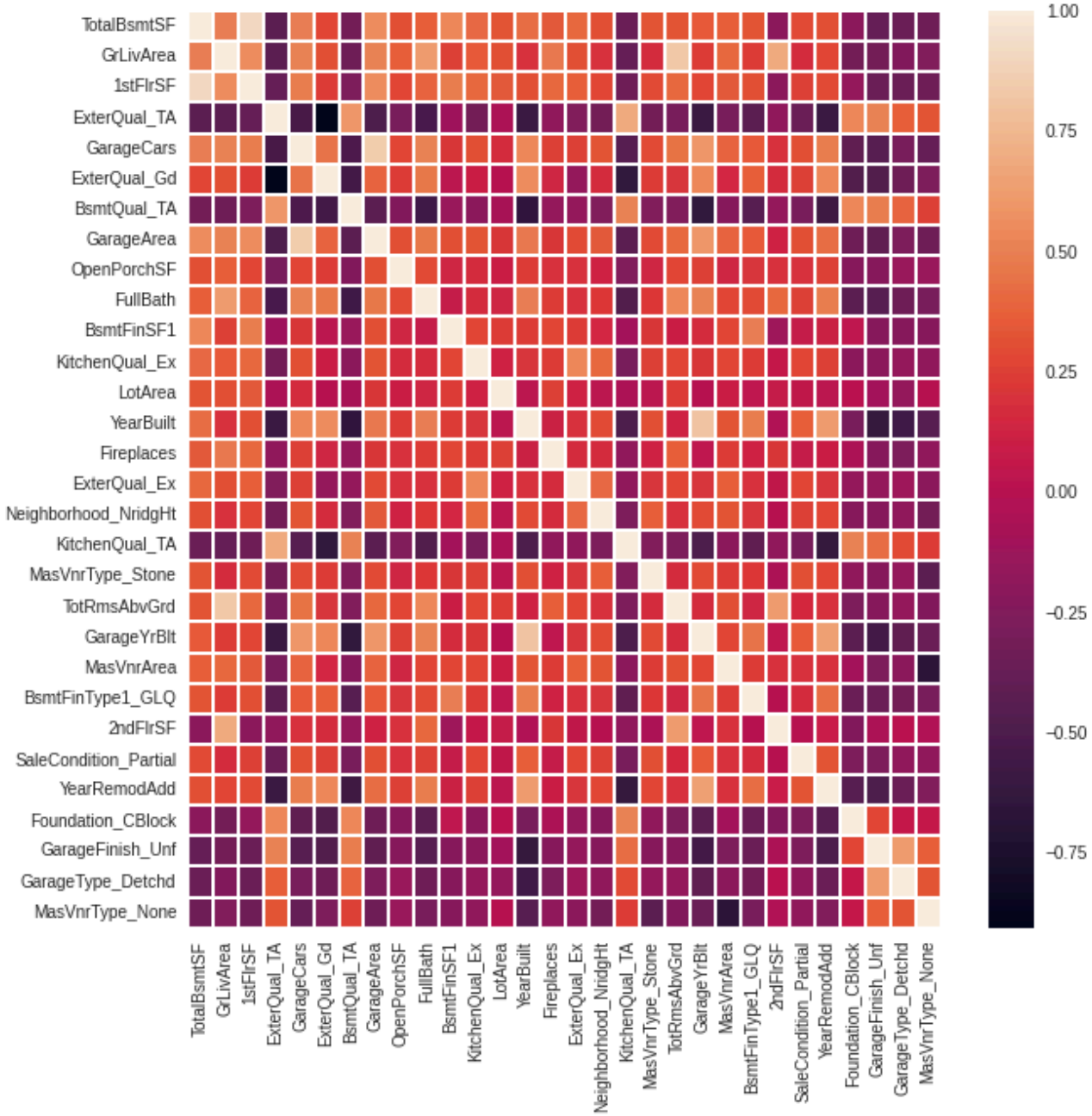
Rozdelenie dát

Naše dáta musia byť rozdelené pre budúce použitie pri trénovaniach a porovnávaniach natrenovaných výsledkov. Preto aby sme vedeli evaluovať úspešnosť naších modelov musíme rozdeliť a vyčleniť stĺpec, ktoré ukazujú hodnoty, ktoré model odhaduje. V našom prípade to je stĺpec **SalePrice**. Taktiež potrebujeme určiť validačnú množinu. Takže dáta sme rozdelili na 2 časti. 70% dát je trenovacích a 30% perecent dát je validačných. Trénovacie sme rozdelili na x_test (všetký stĺpce okrem SalePrice) a y_test (SalePrice).

Korelačná matica

Vďaka koleračnej matici môžeme vizuálne znázorniť závislosti medzi dvoma alebo viacerými premennými. Jednotlivé vzťahy sú podľa prekrývania hodnôt farebne označané. Ak majú dva stĺpce spolu spoločných veľa hodnôt v rovnakých riadkoch tak je štvorec svetlejší. V opačnom prípade štvorec tmavne. Ako môžeme vidieť hlavná diagonála je celá svetlá, čo značí že hodnoty sa prekrývajú. Naše dáta majú príliš veľa stĺpcov nato aby sme ich jednoducho a zretelne zobrazili v koleračnej matici. Preto sme vybrali 25 najdôležitejších stĺpcov pri vyhodnocovaní cenu domu. Tieto stĺpce sme získali po natrenovaní random forest modelu, ktorý si preberieme neskôr. Na koleračnej matici môžeme napríklad vidieť prekryv stĺpcov 1stFloorSF a TotalBsmntSF. Tieto stĺpce zobrazujú rozlohu 2. poschodia a pivnice. Tieto údaje sú poväčšine totožné preto je aj prekryv zvýraznený na bielou farbou. Na druhej strane pri stĺpcoch BsmtQual_TA a BsmtQual_GD vidíme na prekryve tmavu farbu. Ide totiž o one hot encodnute

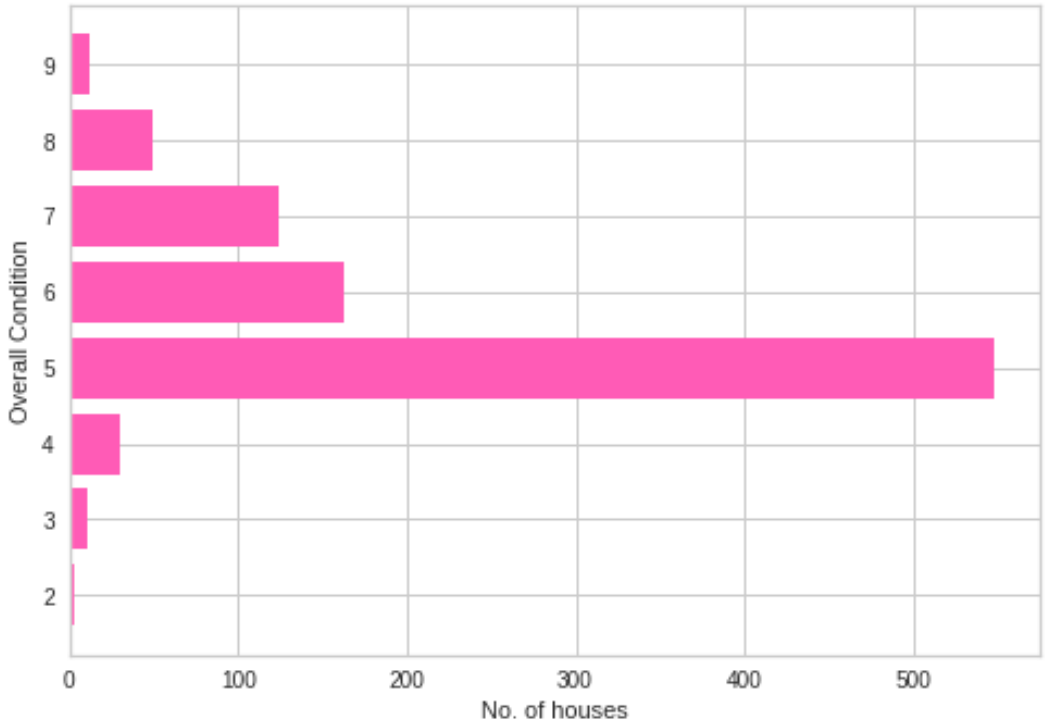
stĺpce, ktoré zobrazujú v akej kondícii je pivnica. Kedže jeden dom nemôže mať dve hodnotenie kondície pivnice, tak prekrývanie hodnôt nikdy nenastane.



Analýza datasetu cez EDA

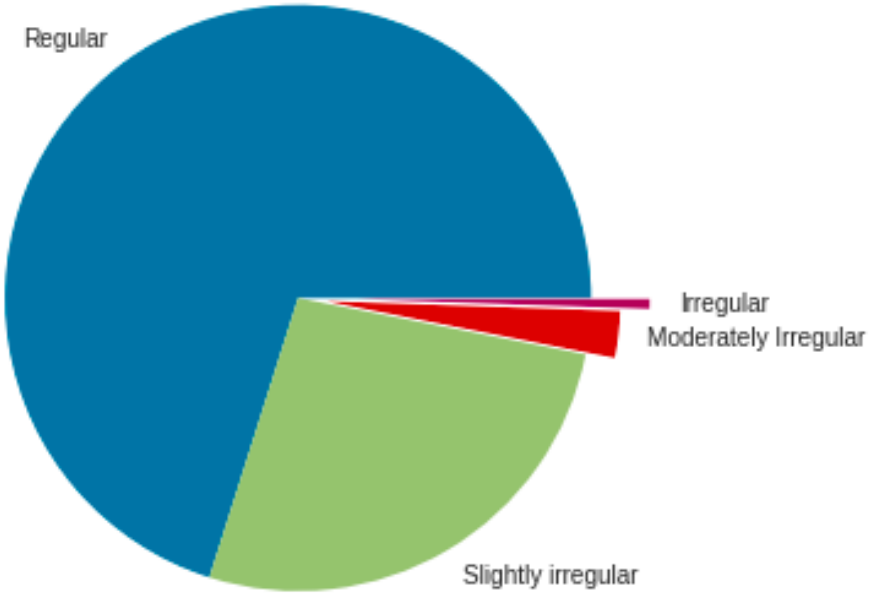
Pre analýzu datasetu sme si pripravili 5 grafov, ktoré zobrazujú súvislosti medzi danými stĺpcami alebo vysvetlenie účelu stĺpca. Na vizualizáciu sme použili histogramy, chart grafy, stĺpcové grafy a koláčový graf.

1. Kondícia domov



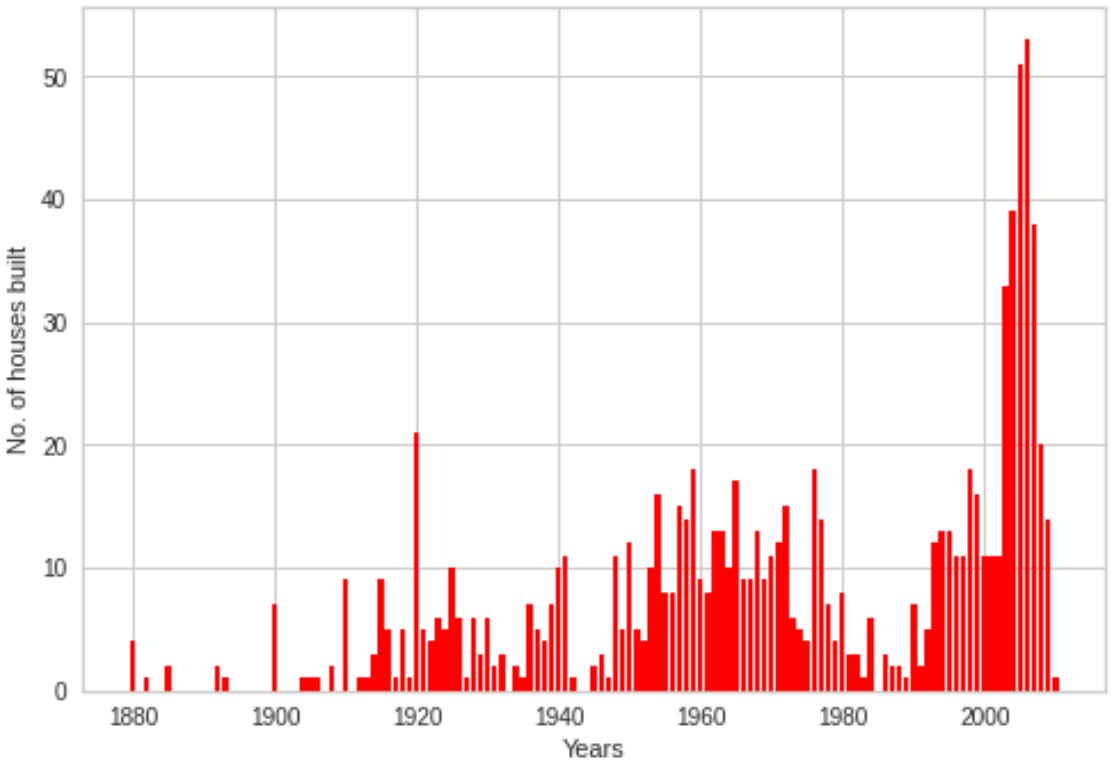
Na stĺpcovom grafe môžeme vidieť počet domov danej kvality. Na osi x vidíme počet domov skupinovaný po 100. Na osi y vidíme kondíciu v akých domy sú. Najviac domov má priemernú/dobrú kondíciu, ktorú reprezentuje číslo 5. Ako môžeme vidieť viac domov je lepšej kvality ako horšej. Napríklad domy s hodnotami od 0 po 2 sa ani v našich dátach nenachádzajú. Je viac domov v perfektnej kondícii ako v najhoršej kondícii.

2. Tvar domov



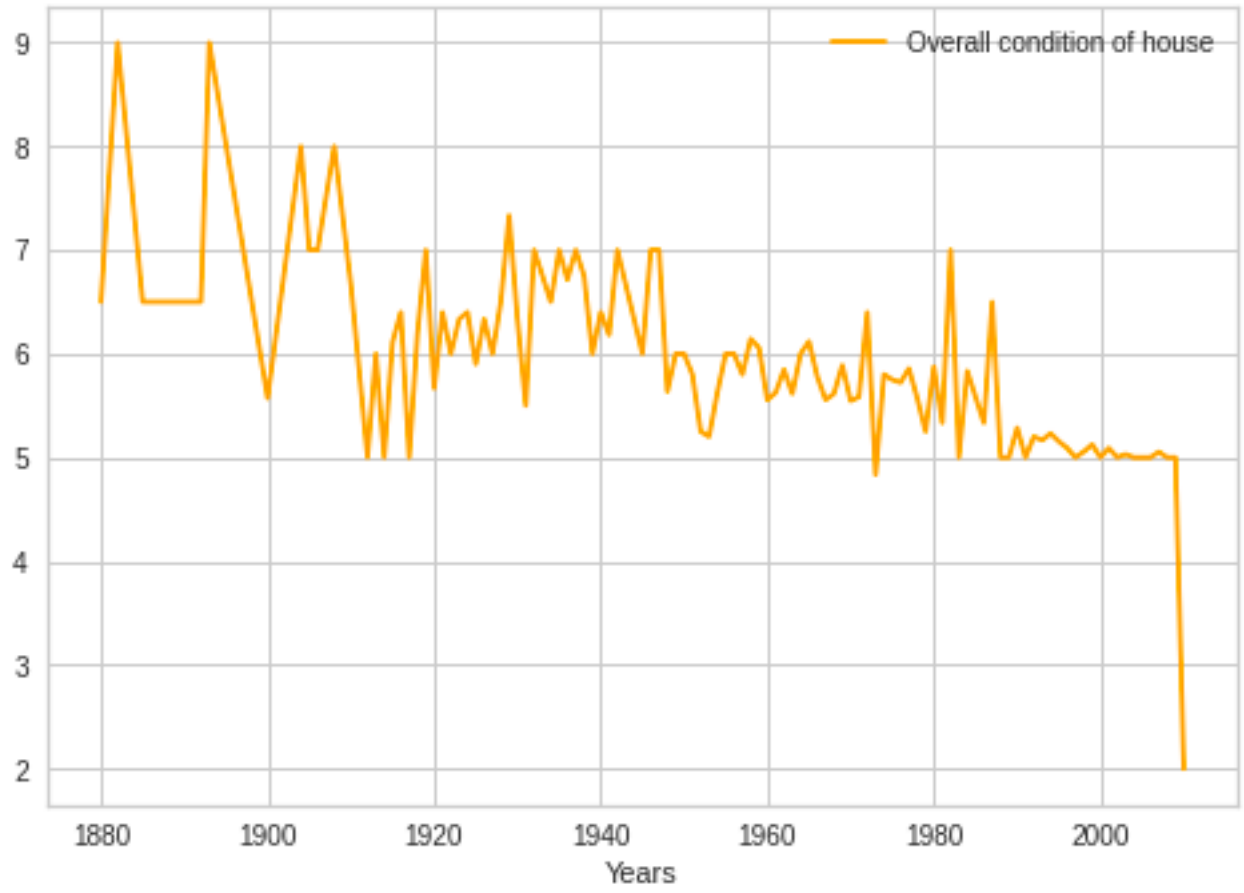
Na koláčovom grafe môžeme vidieť pomer zvyčajných a nezvyčajných tvarov domov. Regular sú domy, ktorý majú obyčajný alebo bežný tvar ako je zauživané. Slightly irregular sú domy, ktoré majú jemné úpravy ale celkovo nepôsobia nezvyčajne. Moderately irregular sú domy, ktoré majú moderný dizajn a zároveň aj moderný tvar, ktorý nie je moc zauživaný. Irregular sú domy, ktoré majú výrazne nezvyčajný tvar, ktoré môžu byť čiastočne aj umeleckým dielom. Ako môžeme vidieť drtivá väčšina domov, cca 70%, v databaze má bežný tvar. Necelá štvrtina má jemné úpravy tvarov. Malá časť domov má značne úpravený vzhľad a zlomok domov má nezvyčajný tvar.

3. Postavené domy počas rokov



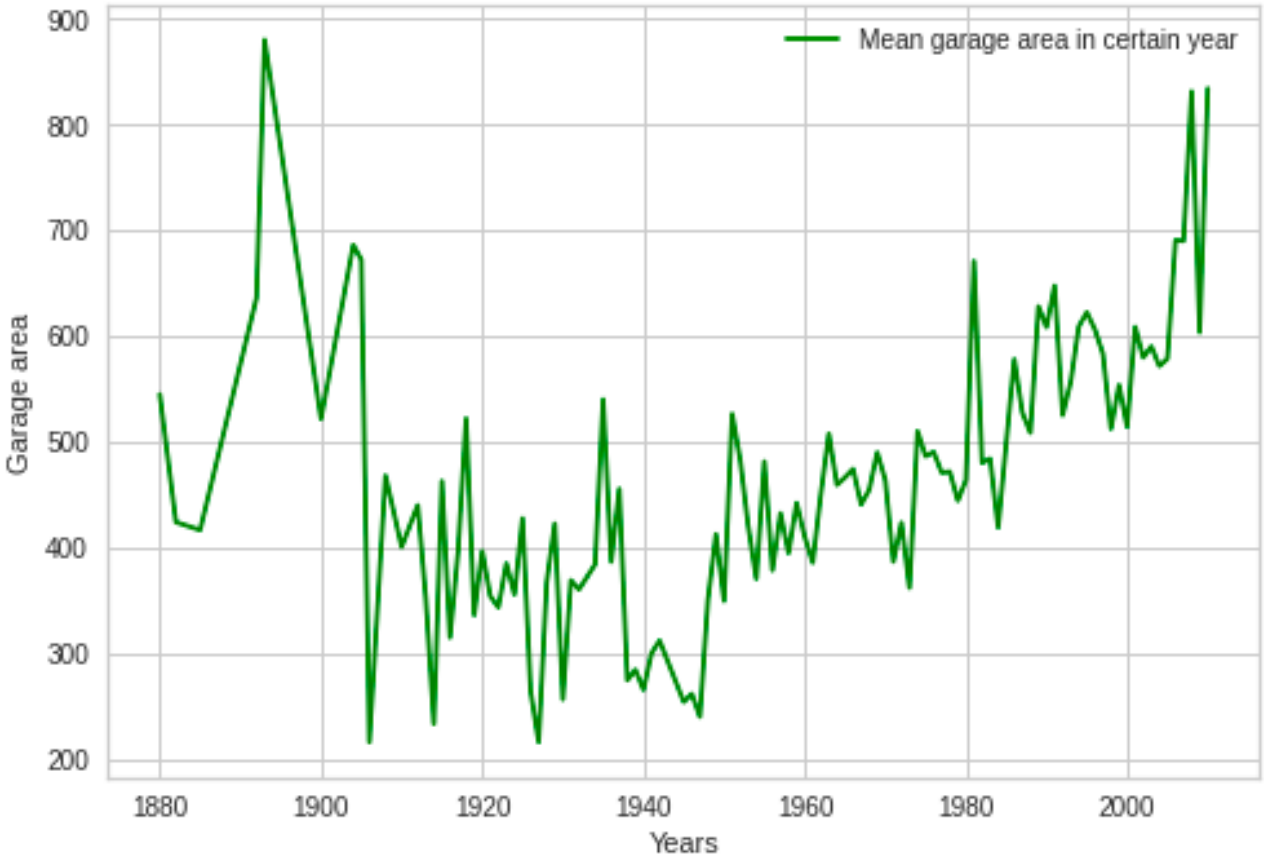
Na grafe môžeme vidieť počet domov postavených počas posledných 140 rokov. Na osi x vidíme roky, ktoré sú skupinované po 20tich rokoch. Na ose y vidíme počet domov postavených za daný rok. Celkový počet domov v našej databáze je 940. Ako môžeme vidieť najvyšší počet domov bolo postavených medzi rokmi 1990 až do roku 2010. Dáta obsahujú aj domy staršie vyše 100 rokov, ale tých je veľmi malé množstvo.

4. Priemerná kondícia domov počas rokov



Na nasledujúcom grafe môžeme vidieť priemernú kvalitu domov za daný rok. Na ose x sú uvedené roky poskupinkované po 2 dekádach. Na ose y môžeme vidieť celkovú kvalitu domov, ktorá sa pohybuje medzi hodnotami 0-9 (0 najhoršia, 9 najlepšia). Prvá zaujímavaosť je, že v databáze sa nenachádzajú takmer žiadne domy s kvalitou menšiou ako 2. Zvláštne na tomto grafe je, že domy staršie než 100 rokov majú v priemere lepšiu kvalitu ako domy postavené v 21. storočí. Na jednej strane, domy, ktoré boli postavené v tom roku sa vyskytujú veľmi zriedkavo. Na druhej strane môžeme poukázať na nekvalitné stavby v 21. storočí. Všetko sa robí čo najrýchlejšie a najlacnejšie a kvalita domov takto výrazne klesá.

5. Vývin priemernej plocha garáže počas rokov



Na grafe mozeme vidiet priemernú veľkosť, v metroch štvorcových, pre dom v daných rokoch. Na osi x zobrazujeme roky a na osi y zobrazujeme veľkosť garáže. V roku 1890 môžeme vidieť vysokú krivku, a to z dôvodu, že v tomto roku sa nachádza zlomkové množstvo domov. Poukázal by som na rast od roku 1950. Postupom do súčasnosti sa zvyšoval počet áut na cestách a taktiež v garážiach. Dnešné rodiny mávajú 2 až 3 autá, čize logicky sa velkosť garáže zväčšovala.

Rozhodovací strom

Prvý model, ktorý sme trénovali bol rozhodovací strom. Trénovanie pomocou rozhodovacieho stromu je výhodne vďaka prehladnosti. Krok po kroku môžeme vidieť ako trénovanie prebiehalo a ako sa rozhodovalo podľa hodnôt v parametroch. Pre výber vhodnýhc parametrov sme použili grid search s cross validáciou. Grid search vyberá najlepšiu kombináciu poskytnutých parametrov pre model podľa presnosti modelu. Našemu modelu sme poskytli nasledujúce parametre.

1. Criterion : funkcia na vyhodnocovanie kvality postupného delenia stromu
2. Max depth : maximálna hĺbka stromu
3. Min sample leafs : minimálny počet prvkov potrebných na existenciu listu v strome
4. Max leaf nodes : maximálny počet potomkov listu
5. Splitter : stratégia pre rozdleovanie listov

Pre parametre sme vybrali nasledovné hodnoty a nechali sme grid search vybrať optimálne parametre.

```
tree_para = {
    'criterion': ["squared_error", "mse", "friedman_mse", "absolute_error"],
    'splitter' : ["best", "random"],
    'max_depth':[4, 6, 8, 10],
    'min_samples_leaf': [10, 20, 40, 80],
    'max_leaf_nodes': [30, 50, 80, 100],
    }
```

```
Best params:  {'criterion': 'friedman_mse', 'max_depth': 8, 'max_leaf_nodes': 50, 'min_samples_leaf': 10, 'splitter': 'random'}
Best scores:  0.746131849963682
Test R2 Score: 0.5689256228407853
Test MSE Score: 0.421896148497049
```

Vo výpise môžeme vidieť najlepšie parametre, najlepšie skóre, r2 skóre a mse skóre. R2 vyjadruje zhody modelu, čiže na 56% náš model správne vyhodnotil údaje podľa testovacích dát. MSE

predstavuje mieru chybovosti na druhú. MSE v tomto prípade je veľmi nízka, čo vyjadruje, že naša siet pomerne úspešne vyhodnotila cenu domu podľa testovacích dát.

Skúsime zmenšiť množinu parametrov na parametre, ktoré sa viac približujú k optimálnym parametrom.

```
tree_para = {
    'criterion': ["squared_error", "mse", "friedman_mse", "absolute_error"],
    'splitter' : ["best", "random"],
    'max_depth':[7, 8, 9],
    'min_samples_leaf': [6, 8, 10, 12],
    'max_leaf_nodes': [40, 50, 60],
    }
```

```
Best params:  {'criterion': 'squared_error', 'max_depth': 8, 'max_leaf_nodes': 40, 'min_samples_leaf': 6, 'splitter': 'best'}
Best scores:  0.7934483864252903
Test R2 Score: 0.5538426371766534
Test MSE Score: 0.43665799447237397
```

Výsledky sa výrazne zlepšili. Nastala zmena pri kriteriálnej funkcii a taktiež z max leaf nodes aj min sample leaf boli vybraté najmenšie hodnoty. Skúsime pre tieto dva parametre vybrať menšie hodnoty.

```
tree_para = {
    'criterion': ["squared_error", "mse", "friedman_mse", "absolute_error"],
    'splitter' : ["best", "random"],
    'max_depth':[7, 8, 9],
    'min_samples_leaf': [2, 4, 6, 8],
    'max_leaf_nodes': [20, 30, 40],
    }
```

```
Best params:  {'criterion': 'friedman_mse', 'max_depth': 8, 'max_leaf_nodes': 40, 'min_samples_leaf': 6, 'splitter': 'best'}
Best scores:  0.7931897326530196
Test R2 Score: 0.5538426371766534
Test MSE Score: 0.43665799447237397
```

Presnosť sa nám nezlepšila, parametre ostali podobné ako pri predošlom výbere.

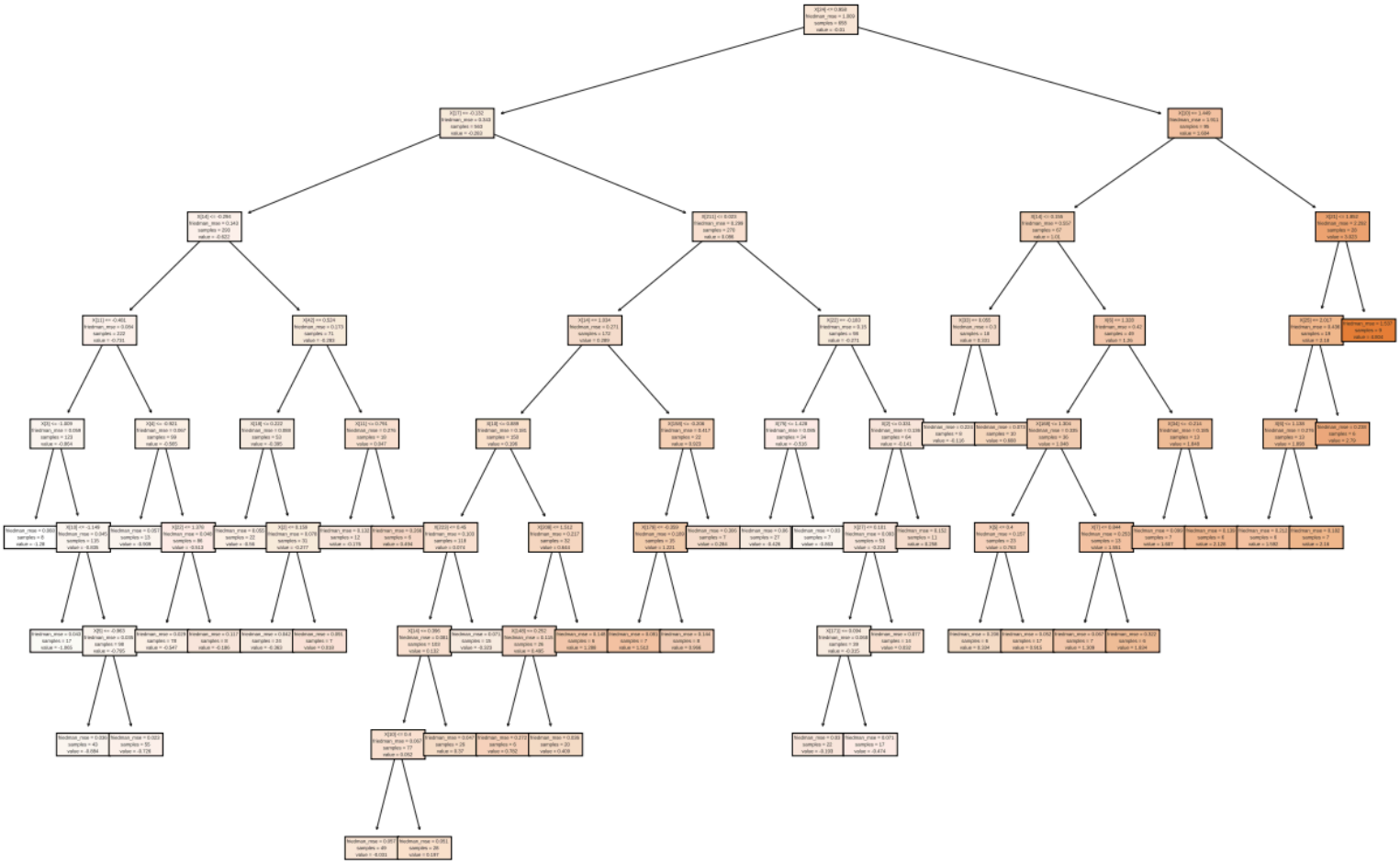
Môžeme teraz experimentovať s cross validáciou. Pri predošlých trénovaniach sme delili trémovaciu množinu na 7 častí pomocou cross validácie. Mohol nastať stav, že sme množinu rozdelili až na veľa častí. Zmenšíme delenie množiny zo 7 častí na 5 častí.

```
Best params:  {'criterion': 'absolute_error', 'max_depth': 9, 'max_leaf_nodes': 40, 'min_samples_leaf': 4, 'splitter': 'best'}
Best scores:  0.7892343181559272
Test R2 Score: 0.591215007249323
Test MSE Score: 0.4000813389592996
```

R2 aj MSE sa nám výrazne zlepšilo. Môžeme to považovať ako najlepší model.

Reziduály zobrazujú presnosť nášho modelu. Stredová horizontálna čiara zobrazuje pravdivú hodnotu. Modré bodky ukazujú predpovedané hodnoty podľa trénovacích dát a zelené bodky podľa testovacích dát.

Na dalšej ukážke môžeme vidieť vizualizáciu rozhodovacieho stromu z najlepšieho modelu.



SVM

Další model, na ktorom sme trénovali naše dáta bol SVM (stroje s podpornými vektormi). SVM kategorizije množiny bodov podľa ich vzájomných vlastností. Pre predstavu SVM kategorizuje body podľa farby do skupín, kde je na grafe jasne vidieť prázdny priestor medzi kategóriami, čiže každý bod má vlastnú skupinu. Prázdny pirestor medzi kategóriami sa dá reprezentovať aj vektormi.

Rovnako ako pri rozhodovacom strome sme použili na výber optimálnych parametrov grid search s cross validáciou. Našemu modelu sme poskytli nasledujúce parametre.

1. Kernel : typ jadra používaný pri algoritme
2. C : regulačný parameter
3. Gamma : koeficient jadra
4. Max iteration : maximálny počet iterácií

Pre parametre sme vybrali nasledujúce hodnoty.

```
svr_para = {
    'kernel': ['rbf', 'poly', 'sigmoid'],
    'C': [0.1,1, 10, 100],
    'gamma': [1,0.1,0.01,0.001]
  }
```

```
Best params:  {'C': 100, 'gamma': 0.001, 'kernel': 'rbf'}
Best scores:  0.8616505195499476
Test R2 Score: 0.78943529392472
Test MSE Score: 0.20608146345418862
```

Výsledky nám vyšli veľmi dobré, napriek tomu skúsime parametre priblížiť k optimálnym hodnotám.

```
svr_para = {
    'kernel': ['rbf', 'poly', 'sigmoid'],
    'C': [70, 100, 150, 200],
    'gamma': [0.1, 0.001, 0.0001]
  }
```

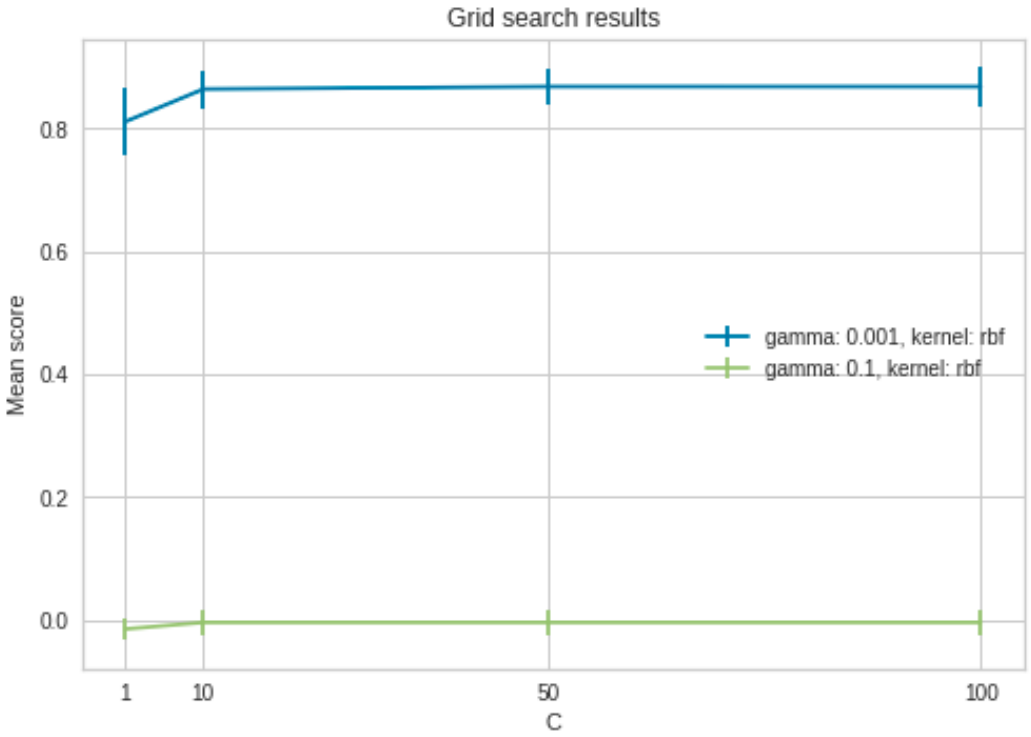
```
Best params:  {'C': 100, 'gamma': 0.0001, 'kernel': 'rbf'}
Best scores:  0.8934917878158417
Test R2 Score: 0.5773089991955315
Test MSE Score: 0.4136912669664488
```

Celkové skóre sa nám zlepšilo ale r2 aj mse chybovosť sa nám výrazne zhoršili. Pri opätovnom spustení s vyššími hodnotami gammi sme sa vrátili k predošlým pozitívnejším hodnotám. Ako posledné otestujeme zvýšiť počet cross validácií z 5 na 7 a upraviť jemne parametre C.

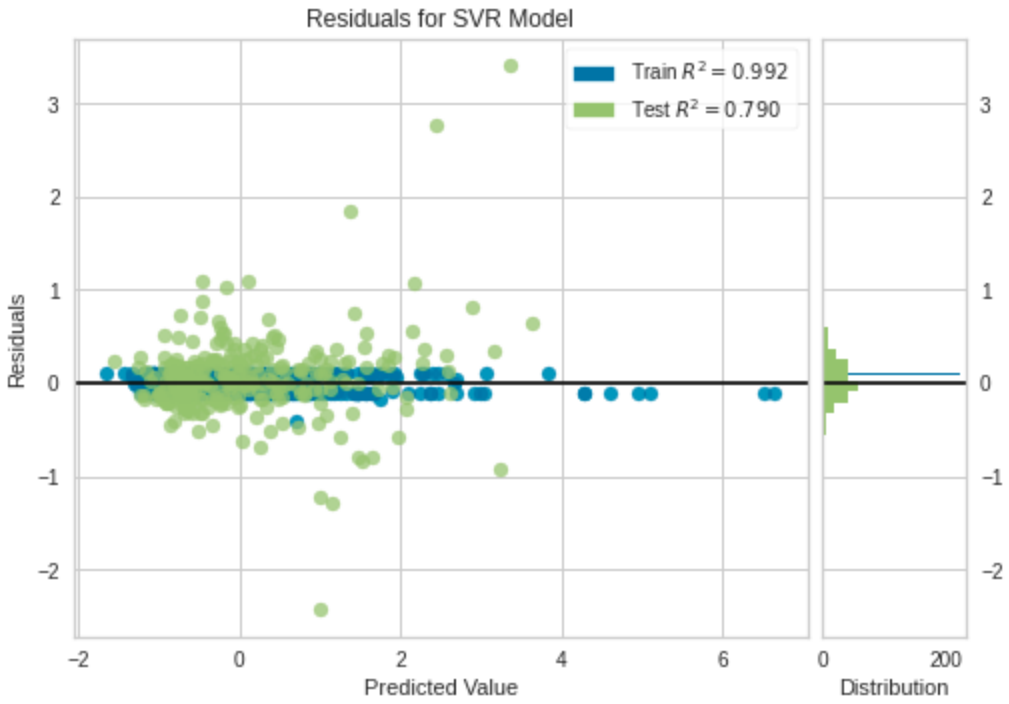
```
Best params:  {'C': 50, 'gamma': 0.001, 'kernel': 'rbf'}
Best scores:  0.8684841246104431
Test R2 Score: 0.7938936432302857
Test MSE Score: 0.20171803918140255
```

Výsledky sa nám mierne zlepšili ale nie o moc. Môžeme to požadovať ako najúspešnejsí model.

Na nasledujúcej ukážke môžeme vidieť vizualizovanú úspešnosť jednotlivých parametrov v grid searchi. Osa y reprezentuje skóre a na osi x máme hodnoty parametru C. Ako moôžeme vidieť spodná čiara, pri ktorej sa párovali hodnoty C s gamma 0.1 a s kernelom rbf sú neúspešné. Naopak pri gamma 0.001 nám vyšli vysoké hodnoty, čiže môžeme považovať gamma parameter za kritický pri vyhodnocovaní.



Následne si zobrazíme reziduály pri najlepšom modeli.



Ako môžeme vidieť odchýlky pru trénovacej množine sú veľmi malé a presné. Na testovacej množine to je o čosi horšie.

Každopádne pri porovnaní presnosti, r2, mse a reziduálov s rozhodovacím stromom je SVR výrazne úspešnejší.

Random forest regressor

Posledným trénovacím modelom je stromový súborový model. Vybrali sme si radnom forest regressor. Trénovanie pomocou súborových modelov je technika, ktorá kombinuje predikcie z rôznych trénovacích algoritmov za účelom čo najlepšieho výsledku. Taktiež sme použili grid search s cross validáciou. Našemu modelu sme poskytli nasledujúce parametre.

1. Bootstrap : použitie bootstrap prvkov
2. Max depth : maximálna hĺbka stromu
3. Max features : maximálny počet stĺpcov
4. Min sample leaf : minimálny počet prvkov s liste
5. Min sample split : minimálny počet delenie prvkov
6. N estimators : počet stromov v “lese”

Použili sme nasledovné parametre pre vyhodnotenie grid searchom.

```
rf_para = {
    'bootstrap': [True],
    'max_depth': [110, 150, 200],
    'max_features': [3, 4],
    'min_samples_leaf': [2, 3],
    'min_samples_split': [6, 8],
    'n_estimators': [50, 100]
}
```

```
Best params:  {'bootstrap': True, 'max_depth': 110, 'max_features': 4, 'min_samples_leaf': 2, 'min_samples_split': 6, 'n_estimators': 100}
Best scores:  0.7332523833858661
Test R2 Score: 0.718477212436081
Test MSE Score: 0.275528739541627
```

Skóre a r2 nám vyšli veľmi podobné. Hodnota r2 je celkom dobrá ale najlepšie skóre by sa dalo jendoznačne zlepšiť. Skúsim upraviť parametre.

```
rf_para = {
    'bootstrap': [True],
    'max_depth': [30, 50, 100],
    'max_features': [4, 5, 8],
    'min_samples_leaf': [1, 2, 3],
    'min_samples_split': [2, 4, 6],
    'n_estimators': [80, 120, 150]
}
```

```
Best params:  {'bootstrap': True, 'max_depth': 100, 'max_features': 8, 'min_samples_leaf': 1, 'min_samples_split': 2, 'n_estimators': 80}
Best scores:  0.8432649396615134
Test R2 Score: 0.7299355045244833
Test MSE Score: 0.2643144119067798
```

Skóre sa po úpravách výrazne zlepšilo. Následne skúsime zmenšiť počet parametrov a priblížiť ich ešte viac k optimálnym hodnotám.

```
rf_para = {
    'bootstrap': [True],
    'max_depth': [100, 150],
    'max_features': [8, 12],
    'min_samples_leaf': [1],
    'min_samples_split': [2],
    'n_estimators': [50, 70, 80]
}
```

```
Best params:  {'bootstrap': True, 'max_depth': 100, 'max_features': 12, 'min_samples_leaf': 1, 'min_samples_split': 2, 'n_estimators': 80}
Best scores:  0.8536511572355
Test R2 Score: 0.7335708748444191
Test MSE Score: 0.26075644414620724
```

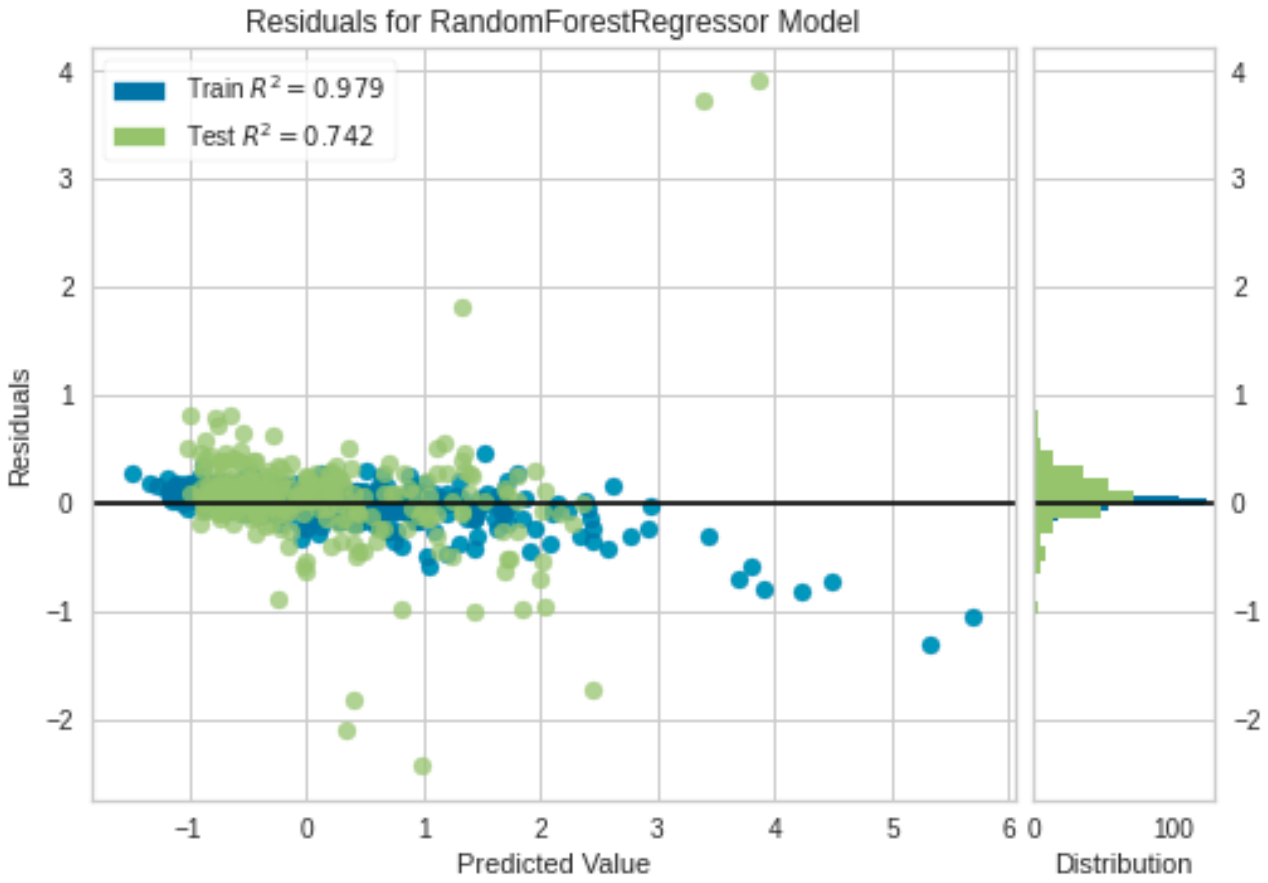
Posledné kolo optimalizácií.

```
rf_para = {
    'bootstrap': [True],
    'max_depth': [90, 100, 110],
    'max_features': [10, 12, 14],
    'min_samples_leaf': [1],
    'min_samples_split': [2],
    'n_estimators': [80, 100, 120]
}
```

```
Best params:  {'bootstrap': True, 'max_depth': 100, 'max_features': 14, 'min_samples_leaf': 1, 'min_samples_split': 2, 'n_estimators': 100}
Best scores:  0.8577753642082815
Test R2 Score: 0.725512732306101
Test MSE Score: 0.26864301658264084
```

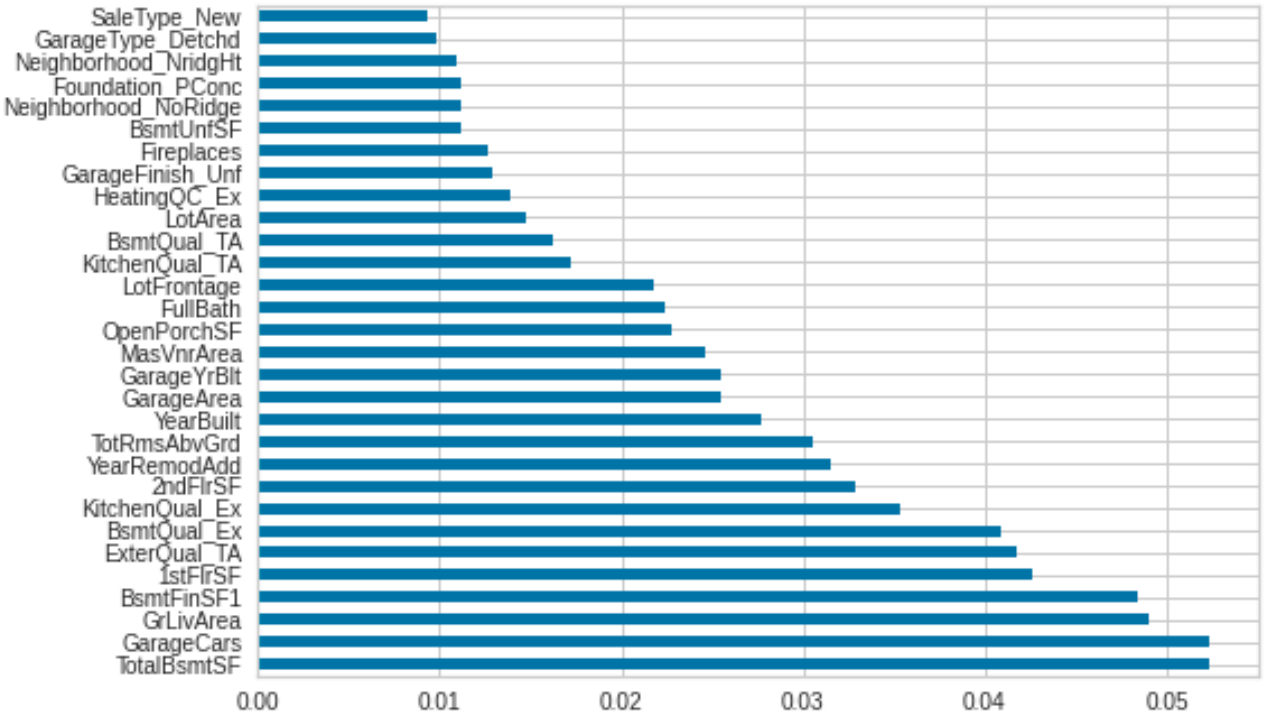
Skóre, r2 aj mse majú už nevýrazne odchýlky, takže môžeme považovať tento model ako najúspešnejší.

Na ukážke môžeme vidieť reziduály.



Hodnoty trénovacej množiny sú perfektné, takmer všetký ležia pri stredovej osi. Hodnoty testvacej množiny majú väčšie odchýlky ale 75% úsešnosť je uspokojivá.

Na stĺpcovom grafe môžeme vidieť dôležistosť najlepších 30 stĺpcov z datasetu.

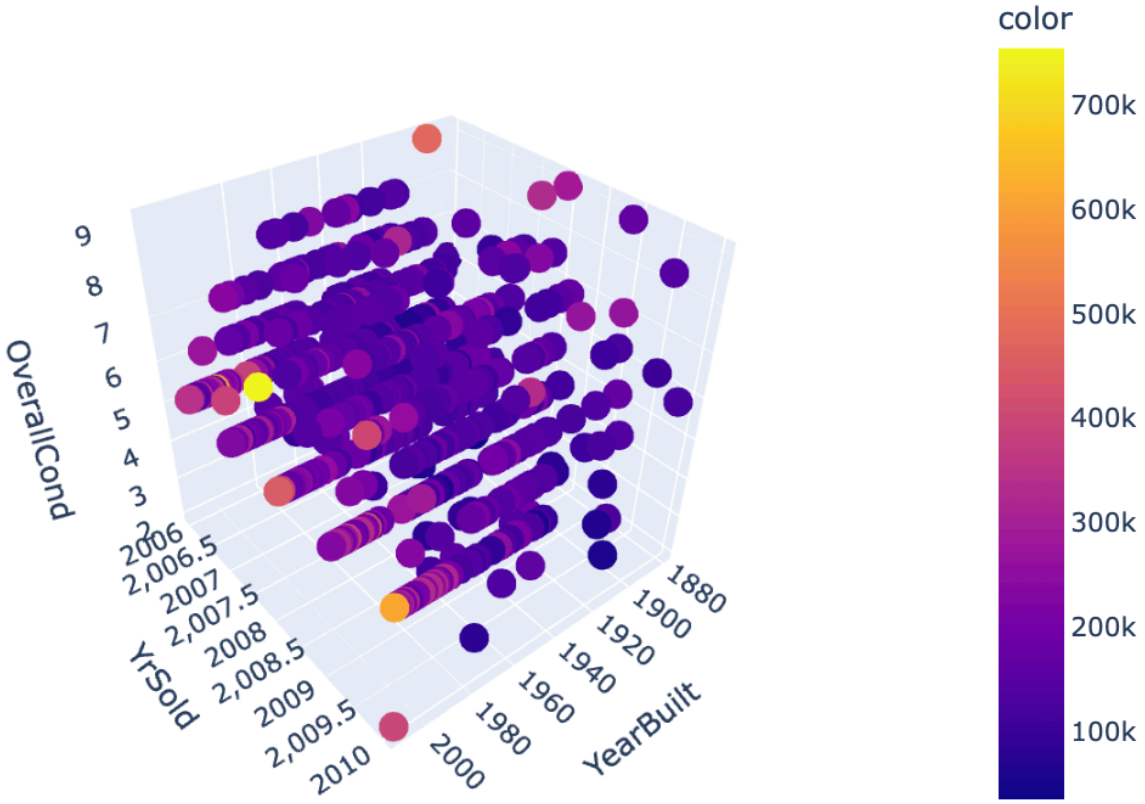


Pri výpočtoch pomocou random forest regressoru najkritickejšími stĺpcami sú celková velkosť podlažia, počet miest pre autá a celková rozloha pozemku. Tieto parametre sú logicky správne, čiže môžeme brať naše výpočty ako pravdivé.

Zo všetkých modelov bol najúspešnejší bol druhý v poradí SVM. Pravdepodobne by najlepšie výsledky mali výjsť v random forest regressore, kedže využiva viacere algoritmi ale našimi parametrami sme to nedokázali. Každopádne random forest a svm mali veľmi podobné výsledky. Pri rozhodovaciom strome nám vyšla najlepšia úspešnosť podobná ale r2 výrazne nižšia a chybovosť pomocou mse nám vyšla v priemere dvojnásobna, preot môžeme požadovať tento model za najmenej úspešný.

Redukcie dimenzie

Analýza príznakov pomocou 3D grafu



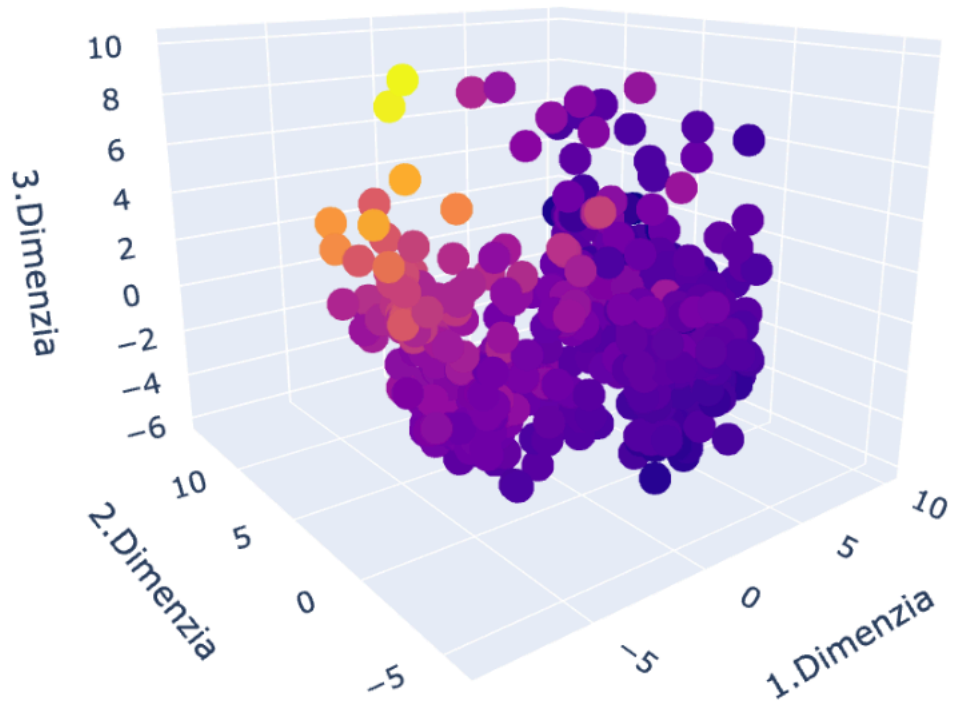
Na 3D grafe môžeme vidieť 3 príznaky. Porovnávame navzájom rok predaja domov, rok postavenie domov a celkovú kondíciu domov. Podľa početnosti bod v grafe žltne, čiže čím viac domov postavených v danom roku a predaných v danom roku s rovnakou kondíciou je tak tým bude bod žltnúť. V opačnom prípade je bod tmavo modrý čo môže reprezentovať napríklad len jeden výskyt. Dáta sme použili bez škálovanie pre lepšiu vizualizáciu. Tento graf sa nám čiastočne prekrýva s našou analýzou datasetu podľa EDA. Ako bolo spomínané, najviac domov má kondíciu hodnotenú číslom 5. Na grafe vidím valcovité tvary po x osi pri

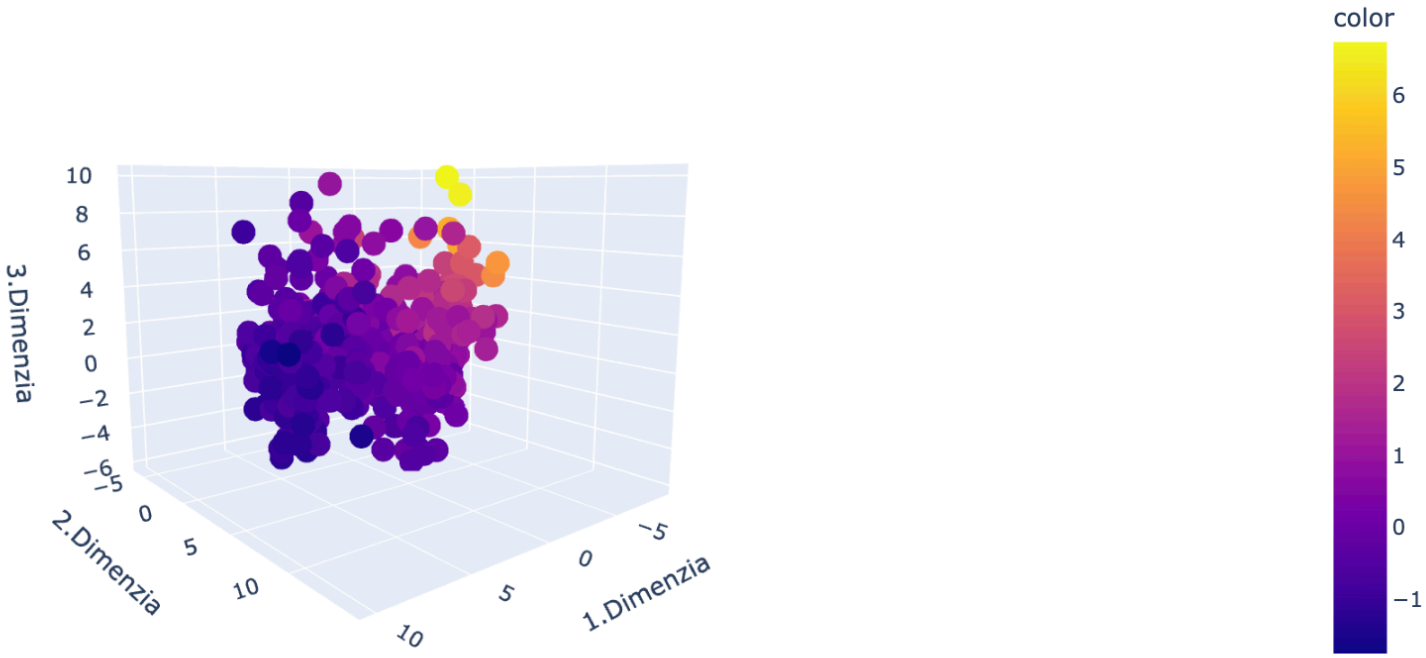
čísle 5. Ďalší poznatok z EDA vieme využiť pri poukázaní nato, že takmer žiaden dom nemá kondíciu nižšiu než 2.

Žltá bodka, reprezentujúca najvyššiu početnosť, sa nachádza pri kondícii číslo 6, v predaných domoch v roku 2007 a postavených okolo roku 1980. Taktiež môžeme vidieť, že mnoho domov postavených po roku 2000 s rokom predaja po roku 2010 má kondíciu hodnotenú číslom 5.

Minimalizovanie množiny na 3 dimenzie

Minimalizovali sme množinu na 3 dimenzie pomocou PCA technológie. PCA nám poskytuje lineárnu redukciu dimenzií podľa zadaného počtu dimenzií. Celú trénovaciu množinu sme zredukovali na 3 dimenzie. Výsledok môžeme vidieť v nasledujúcom 3D grafe



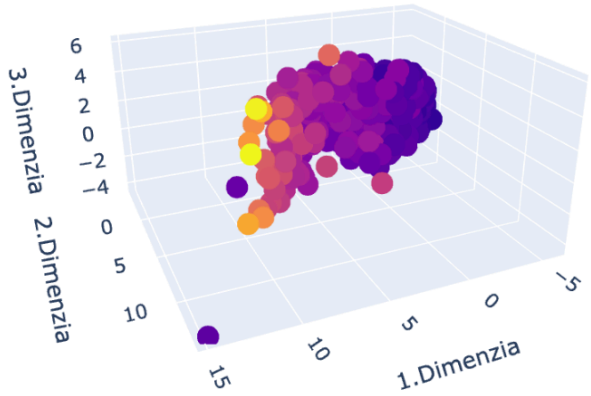


Na grafe môžeme vidieť presnosť modelu pri porovnávaní s príznakom SalePrice. Žlté hodnoty reprezentujú správne vyhodnotenie, tmavo modré presný opak čiže nesprávne klasifikované.

Redukcie dimenzií s trénovaním na najúspešnejšom modeli

Vybrali sme si podmnožinu príznakov na ktorých budeme redukovať dimenzie. Podmnožinu sme vybrali na základe najlepších príznakov z random forest regressora. Pre trénovanie sme použili SVM model s parametrami pre kernel rbf, gammu 0.001 a pre C 50. Tieto parametre mali pre nás navyššiu úspešnosť.

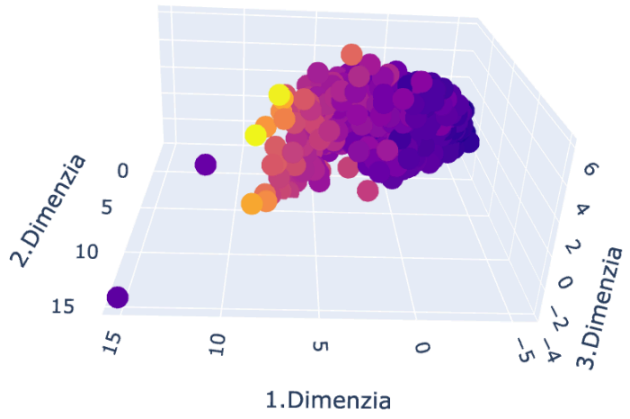
Ako prvé skúsime redukovať na 3 dimenzie. Výsledky sú nasledovné.



Pre redukciu na 4 dimenzie sú výsledky následovné.

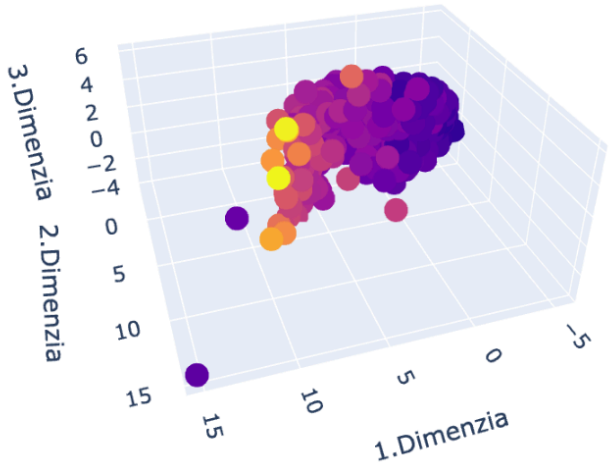
```
Best scores:  0.8433959843952668
Time: 0.0652763843536377
```

Pre 4 dimenzie sú výsledky nasledovné.



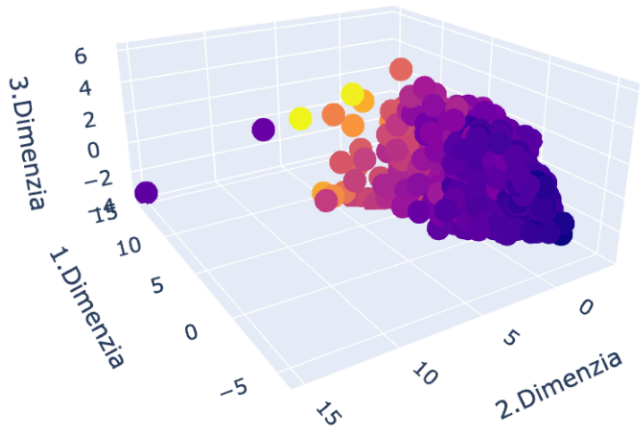

```
Best scores:  0.8526045614778112
Time: 0.06426835060119629
```

Pre 5 dimenzí sú výsledky nasledovné.



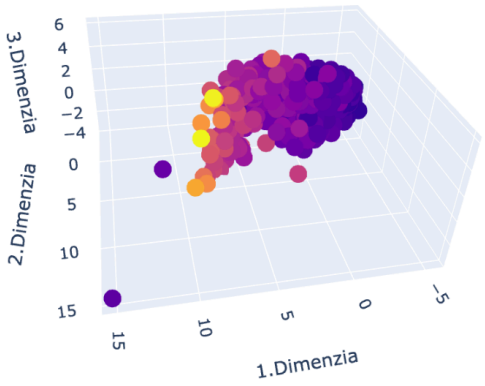

```
Best scores:  0.859139085898449
Time: 0.07254815101623535
```

Pre 6 dimenzí sú výsledky nasledovné.




```
Best scores:  0.8620042264886036
Time: 0.07554435729980469
```

Pre 7 dimenzí sú výsledky nasledovné.




```
Best scores:  0.8630972702731236
Time: 0.06793355941772461
```

Porovananie všetkých výsledkov pri redukcií dimenzií môžeme vidieť na nasledujúcom grafe. Graf zobrazuje vývin času a skóre podľa počtu dimenzií.