Проектна задача по

Веб Програмирање и Вовед во наука за податоци

Тема:

АНАЛИЗА НА ФИНАНСИСКИ ПОДАТОЦИ

Ментор: Ана Тодоровска

Изработиле: Сандра Георгиев 211033

Андреј Влаховски 211136

Филип Аврамоски 211063

Скопје, септември 2024

Содржина

Апстракт

Овој проект се фокусира на анализа на финансиски податоци со комбинирање на Spring Boot и техника на анализа на податоци. Главната цел е да се развијат модели за предвидување и анализа на финансиски трендови и да се прикажат резултатите преку интерактивни визуализации.

Проектот вклучува две главни компоненти: обработка на податоци и визуелизација. Во делот за обработка на податоци, користиме напредни алгоритми за анализа на финансиските податоци, кои вклучуваат импрегнација на податоци, чистење, трансформација и примена на модели за предвидување. Резултатите од анализа се визуелизираат преку интерактивни графики, создадени со Chart.js, и се интегрирани во Spring Boot апликацијата.

Архитектурата на системот е дизајнирана да обезбеди ефикасна обработка на податоци и прегледност на визуелизациите, со цел корисниците да можат лесно да ги анализираат и интерпретираат резултатите. Проектот исто така вклучува методи за тестирање и евалуација на моделите, со акцент на обезбедување точност и релевантност на резултатите.

Проектот демонстрира синергија помеѓу Spring Boot, data science методи и Chart.js, обезбедувајќи целосно решение за анализа и визуелизација на финансиски податоци.

Вовед

Во современиот финансиски сектор, анализата на податоци игра клучна улога во донесувањето на информирани одлуки и предвидување на пазарните трендови. Како резултат на тоа, интеграцијата на напредни аналитички методи со современи технологии за визуелизација стана неопходна за успешна анализа и интерпретација на сложени финансиски податоци.

Овој проект има за цел да ги комбинира предностите на Spring Boot, алатка за развој на веб апликации во Java, со технолошките капацитети на анализа на податоци и визуелизација. Користејќи напредни аналитички техники, проектот ќе развие модели за предвидување на финансиски трендови, кои потоа ќе бидат визуелизирани со помош на Chart.js, библиотека за графички визуализации на JavaScript.

Проектот се состои од две главни фази: обработка на податоци и визуелизација. Во првата фаза, ќе се применат методи за чистење и трансформација на податоците, како и изградба и евалуација на аналитички модели. Втората фаза вклучува интеграција на резултатите од анализата во Spring Boot апликацијата и создавање на интерактивни графики со Chart.js.

Со цел да се постигне висока точност и корисничка искуство, проектот ќе се фокусира на развој на ефикасна архитектура и примена на најдобри практики за интеграција и визуелизација на податоци. Резултатите од проектот ќе помогнат во подобрување на разбирањето на финансиските трендови и поддршка на донесувањето на информирани одлуки.

Овој вовед ја поставува основата за длабочинска анализа и детално разбирање на како технологијата и податоците можат да се комбинираат за постигнување на практични и корисни резултати во финансискиот сектор.

Применети алатки

Во овој проект се користат следните алатки и технологии за развој, обработка на податоци и визуелизација:

Spring Boot

Spring Boot е рамка за развој на Java апликации која овозможува брзо создавање на автономни, продукциски-спремни апликации со минимална конфигурација.

Spring Boot се користи за развој на серверската страна на веб апликацијата. Тоа вклучува управување со логика на апликацијата, интеграција со бази на податоци, и обезбедување на API повици за комуникација со клиентот.

Python

Python е моќен јазик за програмирање со широк спектар на апликации, особено во анализа на податоци и машинско учење.

Python се користи за развој на аналитички модели и алгоритми за обработка на финансиски податоци. Тоа вклучува имплементација на алгоритми за предвидување и анализа на податоци.

Python е широко користен за машинско учење и апликации за вештачка интелигенција поради неговите обемни библиотеки и рамки како што се TensorFlow, Keras и Scikit-learn.

Машинско учење

Машинското учење е област на вештачката интелигенција која се фокусира на развој на алгоритми и модели што овозможуваат компјутерите да учат од податоци и да прават предвидувања или одлуки без експлицитно програмирање. Процесот на машинско учење вклучува собирање и подготовка на податоци, избор и обука на модел, и евалуација на неговата точност и ефективност. Машинското учење има широк спектар на примени, вклучувајќи препораки на производи, откривање на измами, анализа на текст и слика, и многу други области. Технологијата продолжува

да напредува и да се развива, овозможувајќи нови можности и иновации во разни индустрии. Во контекст на анализата на финансиски податоци, машинското учење игра клучна улога во подобрување на точноста на предвидувањата и идентификувањето на значајни образци во податоците. Со помош на машинско учење, можеме да примениме напредни алгоритми за предвидување на финансиски трендови, оценување на ризици, и идентификување на фактори кои влијаат на финансиската стабилност на компаниите.

Scikit-learn

Scikit-learn е Python библиотека за машинско учење која обезбедува широк спектар на алгоритми и алатки за класификација, регресија и кластеризација. Scikit-learn се користи за развој и евалуација на модели за предвидување на финансиски трендови, вклучувајќи примена на алгоритми и анализа на резултати.

XGBoost

XGBoost е напредна библиотека за машинско учење која се фокусира на бустирање на модели за постигнување висока точност и ефикасност. XGBoost се користи за изградба на модели за регресија и класификација со цел да се подобрат резултатите од анализа на податоци.

XGBoost претставува оптимизирана верзија на Gradient boosting кој за разлика од Ada boosting каде составните модели во целосниот модел учат со помош на weighted points, тука за делот на секој составен модел кој имал проблеми се прави нов модел кој се тренира на тој дел од податоците.

Ние во нашиот проект за финансиските податоци користевме XBoostRegressor.

```python
X = df_final.drop('Bankrupt?', axis=1)
y = df_final['Bankrupt?']

model = XGBRegressor(n_estimators=100, random_state=42)
model.fit(X, y)

importances = model.feature_importances_

feature_importances = pd.DataFrame({'Feature': X.columns, 'Importance': importances})
feature_importances = feature_importances.sort_values(by='Importance', ascending=False)

most_important_feature = feature_importances.iloc[0]

print("Most Important Feature:", most_important_feature['Feature'])
print("Importance Score:", most_important_feature['Importance'])

plt.figure(figsize=(14, 8))
sns.barplot(x='Importance', y='Feature', data=feature_importances)
plt.title('Feature Importance for Determining Bankruptcy')
plt.show()
```

Chart.js

Chart.js е библиотека за JavaScript која овозможува креирање на интерактивни и визуелно атрактивни графики.

Chart.js се користи за визуелизација на резултатите од анализата на податоци преку интерактивни графики во веб апликацијата, овозможувајќи на корисниците да ги разгледаат податоците на јасен и интерактивен начин.

Податоци

Во рамките на овој проект, применуваме две различни анализи на податоци за да добиеме целосен увид во финансиските трендови и фактори што влијаат на банкротирање на компании, како и анализи на Fortune 500 компании.

1. Анализа на податоци за 'Bankrupt?'

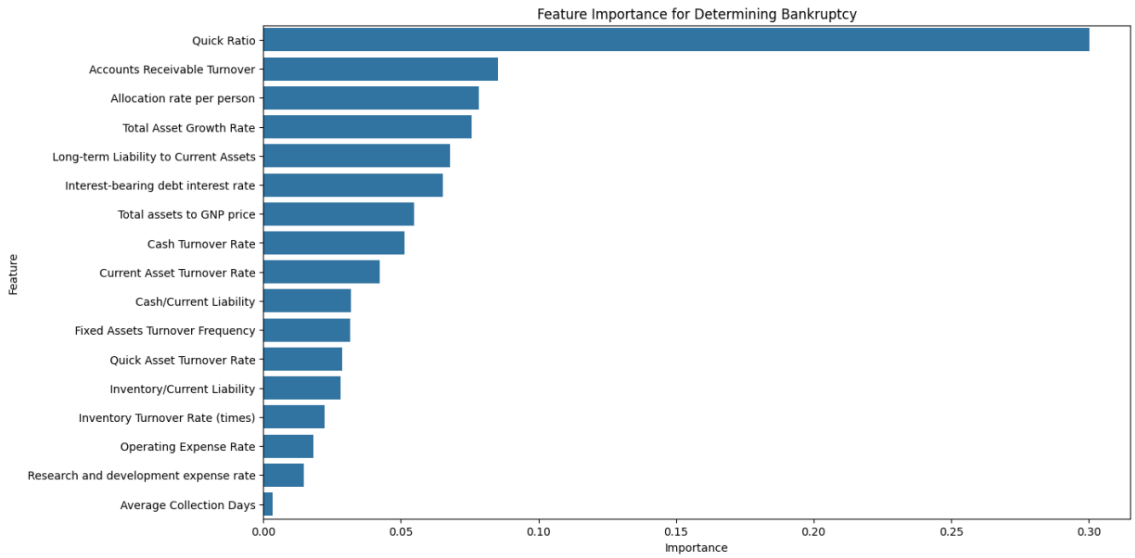
- **Опис на анализа**: Оваа анализа се фокусира на изучување на зависноста помеѓу различните финансиски карактеристики и целната колона 'Bankrupt?'. Целта е да се идентификуваат најзначајните карактеристики кои влијаат на веројатноста за банкротирање на компаниите.

	Bankrupt?	ROA(C) before interest and depreciation before interest	ROA(A) before interest and % after tax	ROA(B) before interest and depreciation after tax	Operating Gross Margin	Realized Sales Gross Margin	Operating Profit Rate	Pre-tax net Interest Rate	After-tax net Interest Rate	Non-industry income and expenditure/revenue	...	Net Income to Total Assets	Total assets to GNP price	No-credit Interval	Gross Profit to Sales	Net Income to Stockholder's Equity	Liability to Equity	Degree of Financial Leverage (DFL)	Interest Coverage Ratio (Interest expense to EBIT)	Net Income Flag	Equity to Liability
0	1	0.370594	0.424389	0.405750	0.601457	0.601457	0.998969	0.796887	0.808809	0.302646	...	0.716845	0.009219	0.622879	0.601453	0.827890	0.290202	0.026601	0.564050	1	0.016469
1	1	0.464291	0.538214	0.516730	0.610235	0.610235	0.998946	0.797380	0.809301	0.303556	...	0.795297	0.008323	0.623652	0.610237	0.839969	0.283846	0.264577	0.570175	1	0.020794
2	1	0.426071	0.499019	0.472295	0.601450	0.601364	0.998857	0.796403	0.808388	0.302035	...	0.774670	0.040003	0.623841	0.601449	0.836774	0.290189	0.026555	0.563706	1	0.016474
3	1	0.399844	0.451265	0.457733	0.583541	0.583541	0.998700	0.796967	0.808966	0.303350	...	0.739555	0.003252	0.622929	0.583538	0.834697	0.281721	0.026697	0.564663	1	0.023982
4	1	0.465022	0.538432	0.522298	0.598783	0.598783	0.998973	0.797366	0.809304	0.303475	...	0.795016	0.003878	0.623521	0.598782	0.839973	0.278514	0.024752	0.575617	1	0.035490
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
6814	0	0.493687	0.539468	0.543230	0.604455	0.604462	0.998992	0.797409	0.809331	0.303510	...	0.799927	0.000466	0.623620	0.604455	0.840359	0.279606	0.027064	0.566193	1	0.029890
6815	0	0.475162	0.538269	0.524172	0.598308	0.598308	0.998992	0.797414	0.809327	0.303520	...	0.799748	0.001959	0.623931	0.598306	0.840306	0.278132	0.027009	0.566018	1	0.038284
6816	0	0.472725	0.533744	0.520638	0.610444	0.610213	0.998984	0.797401	0.809317	0.303512	...	0.797778	0.002840	0.624156	0.610441	0.840138	0.275789	0.026791	0.565158	1	0.097649
6817	0	0.506264	0.559911	0.554045	0.607850	0.607850	0.999074	0.797500	0.809399	0.303498	...	0.811808	0.002837	0.623957	0.607846	0.841084	0.277547	0.026822	0.565302	1	0.044009
6818	0	0.493053	0.570105	0.549548	0.627409	0.627409	0.998080	0.801987	0.813800	0.313415	...	0.815956	0.000707	0.626680	0.627408	0.841019	0.275114	0.026793	0.565167	1	0.233902

6819 rows × 96 columns

- **Методи**:
    - **Feature Importance**: Извлекуваме значење на карактеристиките користејќи различни методи, како што се RandomForestClassifier и XGBoost. Ова ни помага да разбереме кои карактеристики имаат најголемо влијание врз предвидувањето на банкротирање.

```
X = df_final.drop('Bankrupt?', axis=1)
y = df_final['Bankrupt?']

model = XGBRegressor(n_estimators=100, random_state=42)
model.fit(X, y)

importances = model.feature_importances_

feature_importances = pd.DataFrame({'Feature': X.columns, 'Importance': importances})
feature_importances = feature_importances.sort_values(by='Importance', ascending=False)

most_important_feature = feature_importances.iloc[0]

print("Most Important Feature:", most_important_feature['Feature'])
print("Importance Score:", most_important_feature['Importance'])

plt.figure(figsize=(14, 8))
sns.barplot(x='Importance', y='Feature', data=feature_importances)
plt.title('Feature Importance for Determining Bankruptcy')
plt.show()
```



2. Анализа на интеракцијата на технолошките компаниите и пазарот на стоки

- **Опис на анализа**: Втората анализа се фокусира на компании, анализирајќи финансиски податоци и идентификувајќи трендови и шаблони во нивната финансиска стабилност и перформанси.
- **Методи**:
    - **Статистичка анализа**: Применуваме статистички методи за анализа на распределба на финансиски показатели, трендови и корелации помеѓу различни финансиски метрики.
    - **Визуелизација**: Креираме графики и визуелизации со Chart.js за да ги претставиме резултатите од анализа, вклучувајќи трендови, распределби и корелации помеѓу финансиските показатели

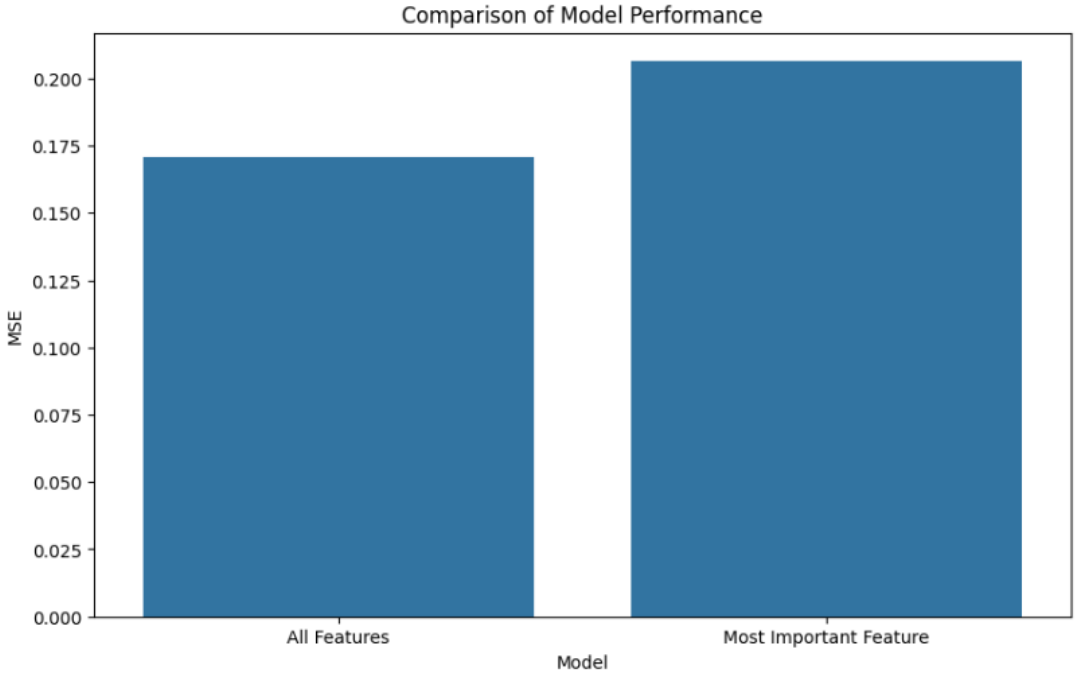
Овие анализи се клучни за разбирање на факторите што влијаат на финансиската стабилност на компаниите и за обезбедување на корисни информации за донесување на информирани одлуки во финансискиот сектор. Користените алатки и методи помагаат да се извлечат значајни увиди и да се прикажат резултатите на јасен и интерактивен начин.

Визуелизација на податоци

Во нашиот проект, визуелизацијата игра важна улога во претставувањето на резултатите од анализа на финансиските податоци и во споредбата на моделите за предвидување. Се користат различни техники и алатки за да се добие јасен увид во ефективноста на моделите и значењето на карактеристиките.

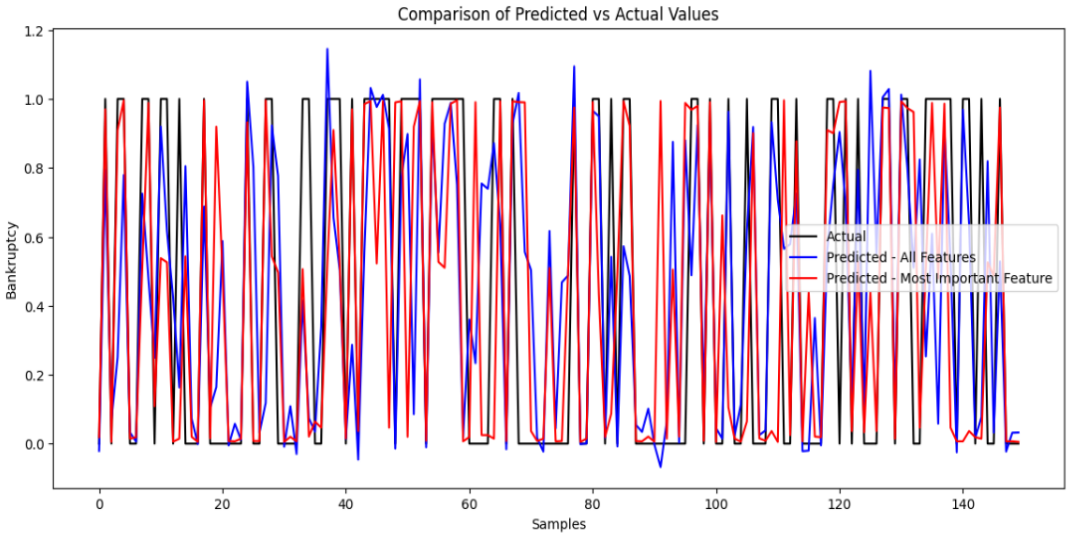
Повеќекратна визуелизација на моделите:

- Споредба на MSE (Средна Квадратна Грешка): За да се процени перформансата на моделите, креиравме бар график со помош на matplotlib и seaborn, кој ја прикажува споредбата помеѓу моделот користејќи сите карактеристики и моделот користејќи само најзначајната карактеристика. Овој график ја илустрира разликата во точноста на моделите, со што се дава визуелен преглед на тоа кој модел е поефикасен.

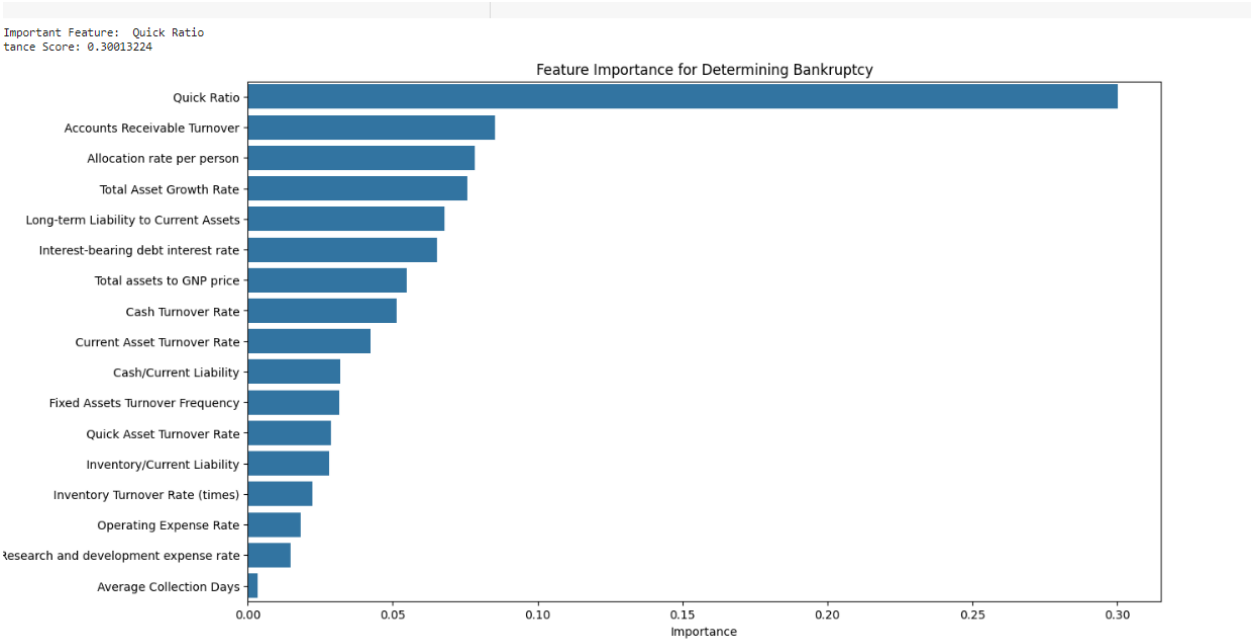


- Споредба на Прогнозирани и Вистински Вредности: Креиравме линиски график кој ја споредува прогнозата на моделот со сите карактеристики и прогнозата на моделот со најзначајната карактеристика со реалните

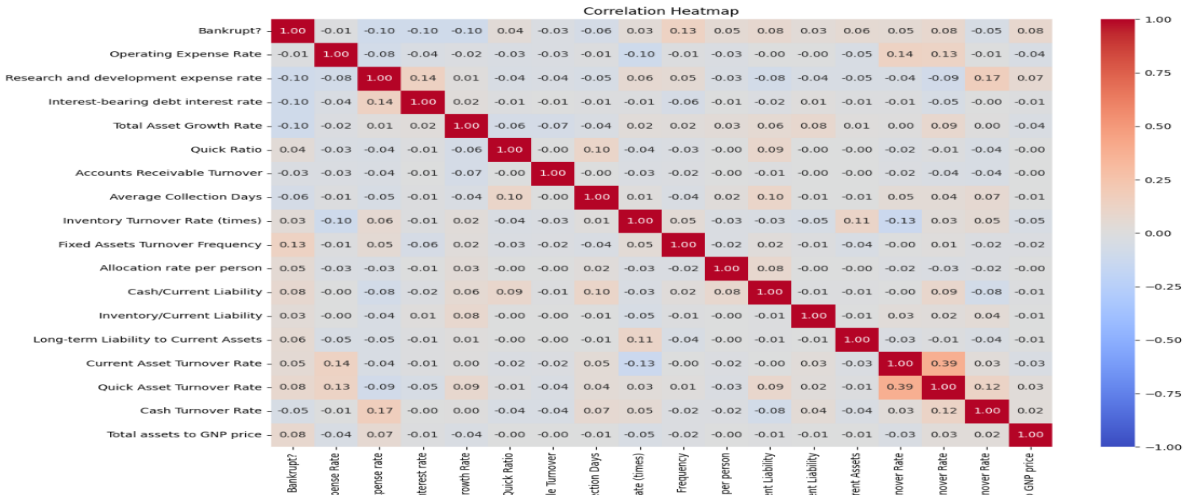
вредности од тест сетот. Овој график помага да се визуелизира колку точно моделите предвидуваат вредности во споредба со вистинските податоци.



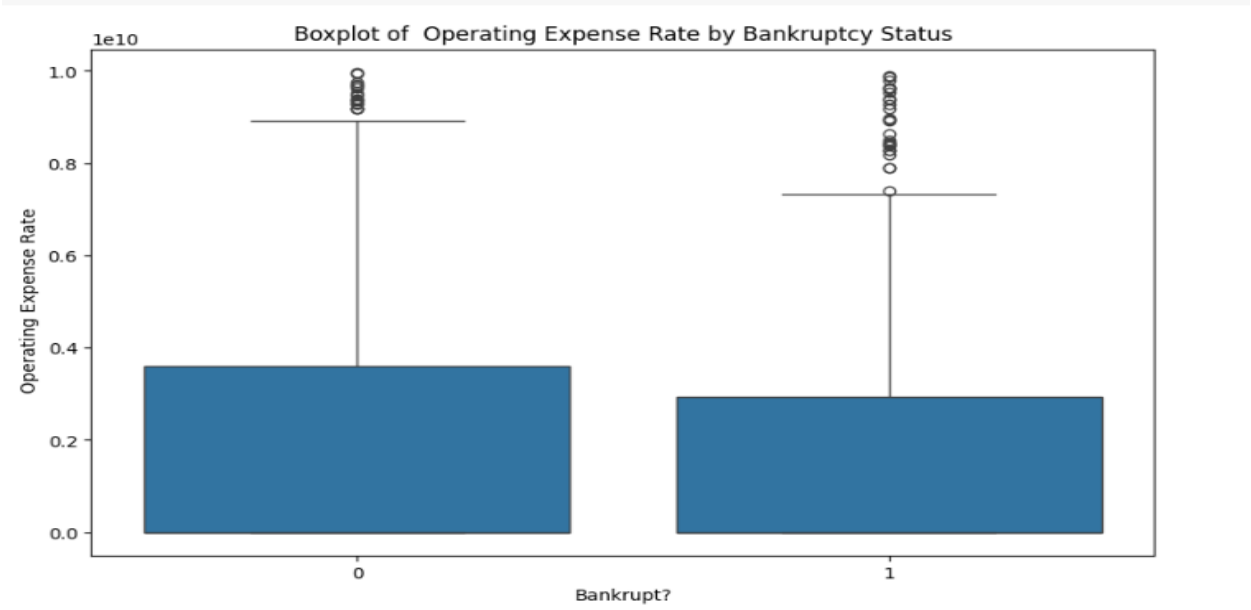
- График на Важност на Карактеристики: Применивме методи за важност на карактеристиките за да ја идентификуваме најзначајната карактеристика која најмногу влијае врз целната колона 'Bankrupt?'. Овие резултати се прикажани во бар график, кој визуелизира важноста на секоја карактеристика, овозможувајќи ни да видиме кои карактеристики се најзначајни за моделот.



- Correlation Heatmap: Креиравме топла мапа на корелација за да ја прикажеме корелацијата помеѓу различните финансиски карактеристики. Ова помага да се идентификуваат врските помеѓу променливите и да се разберат можните взаемни влијанија помеѓу нив.



- Boxplot: Применивме boxplot графици за да ја прикажеме распределбата на различни финансиски карактеристики во зависност од статусот на банкротирање. Овие графици визуелизираат како се распределуваат вредностите на карактеристиките за компании кои се банкротирани и оние кои не се, што помага да се идентификуваат потенцијални разлики и шаблони во податоците.

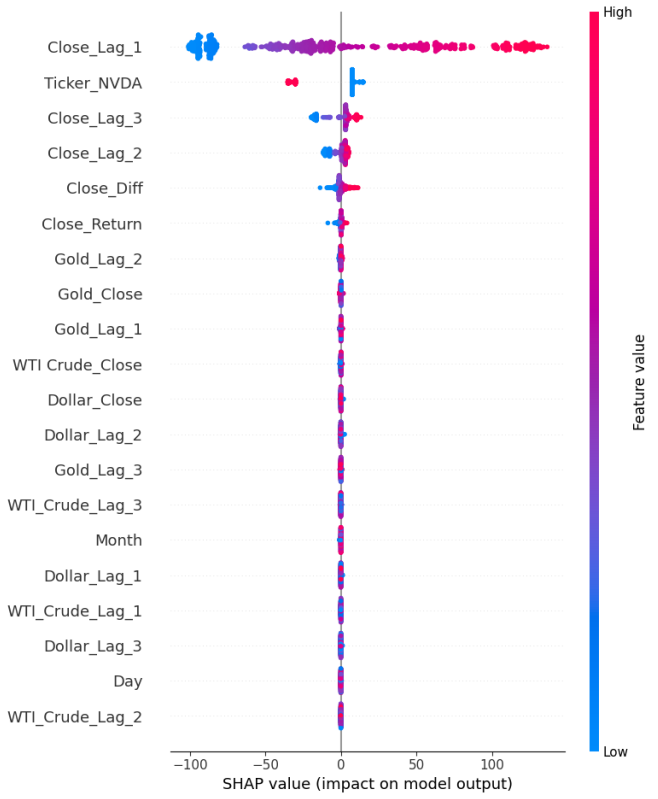


Сите овие визуелизации овозможуваат подлабочен увид во резултатите од нашата анализа и помагаат во донесувањето на информирани одлуки базирани на податоците.

SHAP (Shapley Additive exPlanations модел)

SHAP (Shapley Additive exPlanations) е методологија која обезбедува длабинско разбирање на моделите за машинско учење. SHAP ги разложува предвидувањата на моделите врз основа на придонесот на секоја карактеристика, овозможувајќи транспарентност за тоа како секој влез влијае на конечните предвидувања на моделот. Овој пристап нуди фер и конзистентно мерење на важноста на карактеристиките во различни модели, што го прави особено корисен во сложени анализи како што е предвидувањето на акциите на компаниите, што е во согласност со темата што ја истражуваме. Примената на SHAP овозможува детално и јасно толкување на сложените врски меѓу променливите во предвидувањата. SHAP TreeExplainer е адаптиран за XGBoost моделот, со што се обезбедува длабинско разбирање за тоа како секоја карактеристика влијае на предвидувањата на моделот.

Графиконите на SHAP обезбедуваат визуелна претстава за важноста на карактеристиките, нагласувајќи ја големината и насоката на влијанието на секоја влезна карактеристика врз предвидувањата на моделот. Со визуелизирање на вредностите на SHAP, се добива јасна слика на факторите кои значително влијаат на предвидените промени на цените на храната.



Заклучок

Проектот за анализа на финансиски податоци, кој вклучува примена на машинско учење и визуелизација, обезбеди значајни увидувања и резултати кои се клучни за разбирање на факторите што влијаат на банкротирање на компании и анализа на технолошките компании и интеракцијата со пазарната стока. Примената на машинско учење, особено преку модели како XGBoost, ни овозможи да идентификуваме и анализираме најзначајните карактеристики што имаат најголемо влијание на веројатноста за банкротирање. Овие модели помагаат во подобрувањето на точноста на прогнозите и во проценката на ризици.

Визуелизацијата на податоците преку различни графики и топли мапи на корелација не само што ги прикажува резултатите на моделите, туку и овозможува длабочински увид во распределбата на финансиските карактеристики и нивните врски. Споредбата на моделите користејќи сите карактеристики против само најзначајната карактеристика покажува дека иако најзначајната карактеристика може да обезбеди добро приближување, користењето на сите карактеристики често води до подобра точност и поцелосно разбирање на податоците.

Сите резултати и визуелизации ја потврдуваат важноста на внимателно избирање на карактеристиките и примената на напредни методи за машинско учење за подобрување на финансиската анализа. Проектот исто така демонстрира како комбинирањето на овие техники може да се применат за создавање на моќни и информирачи алатки за анализа и предвидување во финансискиот сектор. Овие увиди и алатки ќе помогнат за донесување на поинформирани одлуки и за поефективно управување со финансиските ризици.