

TUGAS PROGRAM STUDI INDEPENDEN

ORBIT FUTURE ACADEMY

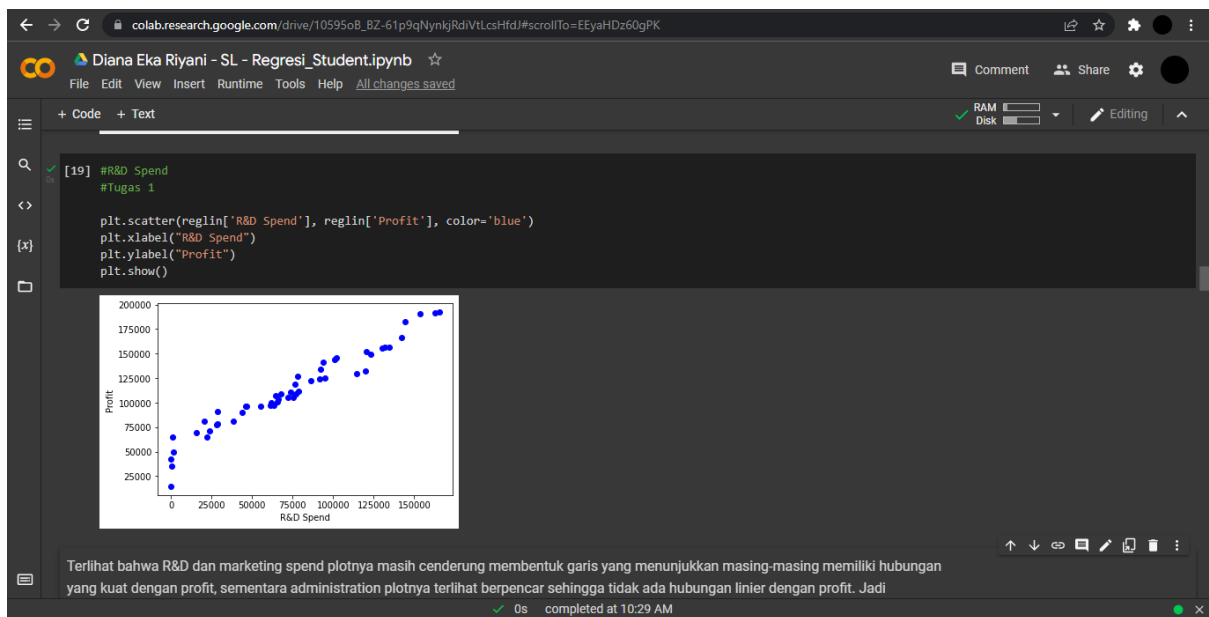
Identitas

1. Nama : Diana Eka Riyani
2. NIM : 24010119140041
3. Asal Kampus : Universitas Diponegoro
4. Kelas : Jupyter XXI
5. Coach : Ipin Sugiyarto
6. Program : Foundations of AI and Life Skills for Gen-Z
7. Hari, Tanggal : Selasa, 15 Maret 2022

Coding SL Regresi

Tugas 1: Mencoba Melakukan Scatter Plot untuk Administration dan R&D Spend

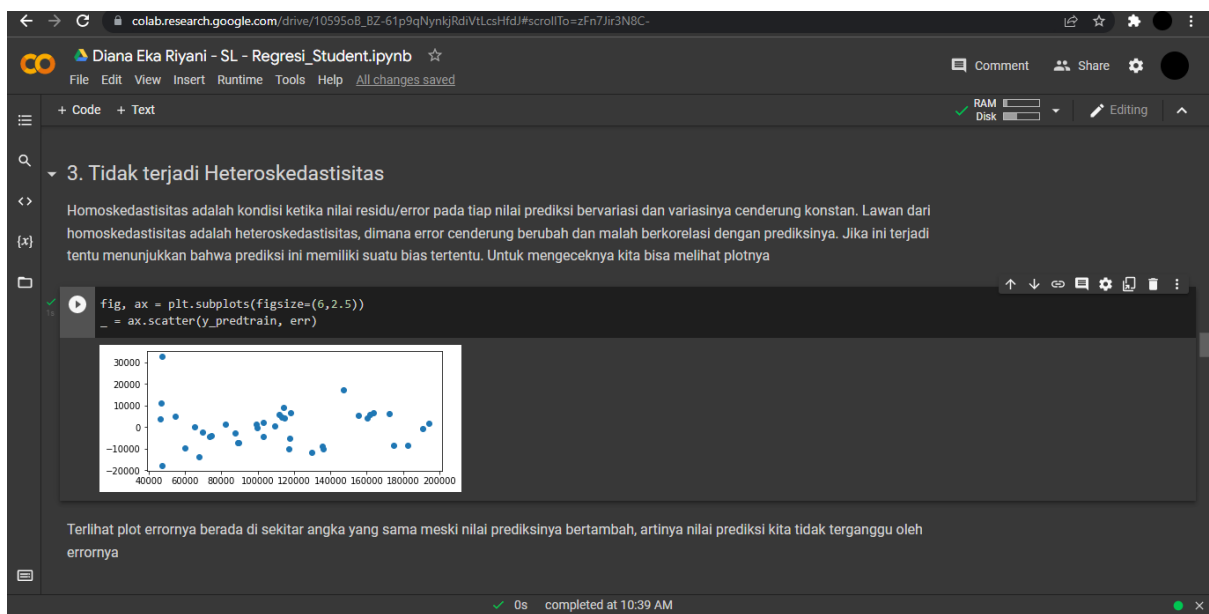
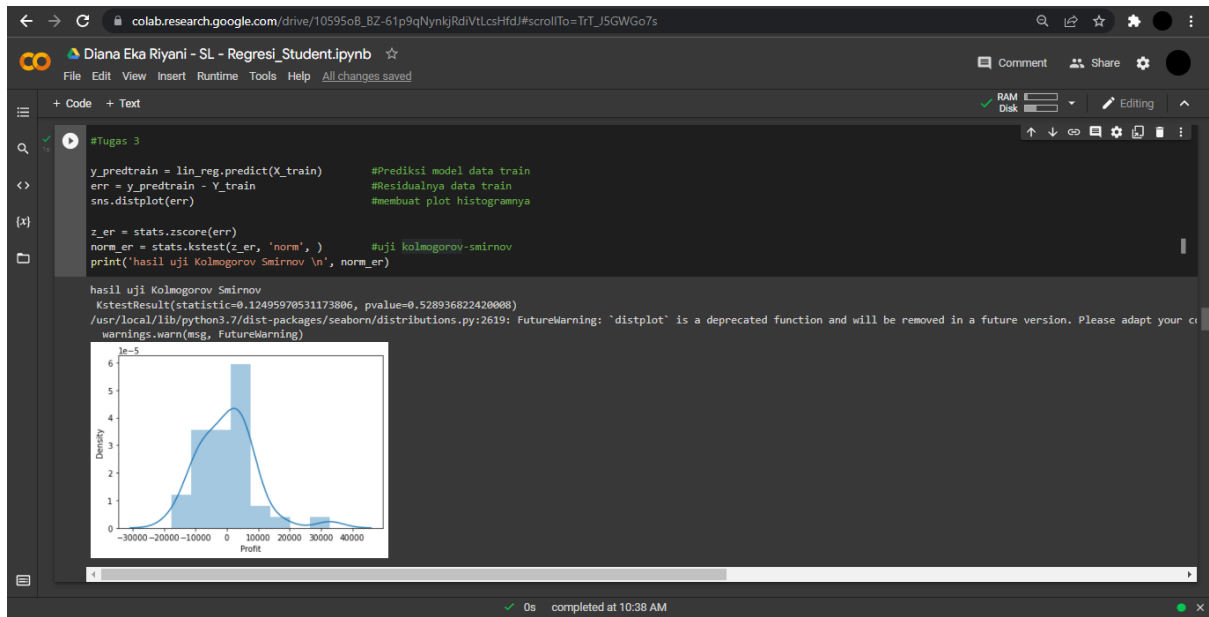




Terlihat bahwa R&D dan marketing spend plotnya masih cenderung membentuk garis yang menunjukkan masing-masing memiliki hubungan yang kuat dengan profit. Sementara administration plotnya terlihat berpecah sehingga tidak ada hubungan linier dengan profit. Jadi, kedepannya variabel yang digunakan sebagai feature adalah R&D dan marketing spend.

Tugas 2: Melakukan Splitting Data

Tugas 3: Uji Normalitas



colab.research.google.com/drive/10595oB_BZ-61p9qNynkjRdiVtLcsHfdJ#scrollTo=c2UKr6rjBDM5

Diana Eka Riyani - SL - Regresi_Student.ipynb

File Edit View Insert Runtime Tools Help

+ Code + Text

4. Tidak terjadi multikolinearitas (regresi berganda)

Multikolinearitas maksudnya adalah hubungan yang kuat antar feature. Regresi linier mengasumsikan bahwa feature-featurenya tidak saling berhubungan. Tentu saja ini hanya berlaku untuk regresi linier berganda (yang featurenya lebih dari 1). Salah 1 cara mengujinya adalah nilai VIF.

```
vif = [variance_inflation_factor(X_train, i) for i in range(len(X_train.T))]
pd.DataFrame({'VIF': vif[0:]}, index=features).T
```

	R&D Spend	Marketing Spend
VIF	6.637335	6.637335

Some papers argue that a $VIF < 10$ is acceptable, but others says that the limit value is 5.

- *10" as the maximum level of VIF (Hair et al., 1995)
- *5" as the maximum level of VIF (Ringle et al., 2015)

Jadi berdasarkan kriteria Hair tidak terjadi kolinearitas, tetapi kalo berdasarkan kriteria Ringle masih terjadi kolinearitas. Disini kita menggunakan teori Hair.

colab.research.google.com/drive/10595oB_BZ-61p9qNynkjRdiVtLcsHfdJ#scrollTo=bo2k_9XORW3p

Diana Eka Riyani - SL - Regresi_Student.ipynb

File Edit View Insert Runtime Tools Help All changes saved

+ Code + Text

5. Tidak terjadi autokorelasi

Autokorelasi adalah hubungan yang erat antar entry, misalnya antara data ke 4 dengan ke 5, data ke-6 dengan ke-7, dll. Ini juga harus dihindari dalam regresi linier. Uji ini tersedia di library yang lain, sehingga kita perlu training model lagi menggunakan library itu.

```
X_constant = sm.add_constant(X_train)
linreg = sm.OLS(Y_train, X_constant).fit()
linreg.summary()
```

OLS Regression Results

Dep. Variable:	Profit	R-squared:	0.957
Model:	OLS	Adj. R-squared:	0.954
Method:	Least Squares	F-statistic:	488.7
Date:	Tue, 15 Mar 2022	Prob (F-statistic):	5.97e-26
Time:	03:41:09	Log-Likelihood:	-421.18
No. Observations:	40	AIC:	848.4
Df Residuals:	37	BIC:	853.4
Df Model:	2		
Covariance Type:	nonrobust		
	coef	std err	t
const	4.639e+04	2887.411	16.068
x1	0.8251	0.045	18.249
x2	0.0236	0.017	1.378
Omnibus:	13.188	Durbin-Watson:	2.147
Prob(Omnibus):	0.001	Jarque-Bera (JB):	16.774
Skew:	-0.984	Prob(JB):	0.000228
Kurtosis:	5.520	Cond. No.	4.98e+05

Warnings:

[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.

[2] The condition number is large, 4.98e+05. This might indicate that there are strong multicollinearity or other numerical problems.

0s completed at 10:41 AM

Tugas 4: Evaluasi Model

```
colab.research.google.com/drive/10595oB_BZ-61p9qNynkjRdVtLcsHfdJ#scrollTo=8RitAMStfVWL
Diana Eka Riyani - SL - Regresi_Student.ipynb
File Edit View Insert Runtime Tools Help All changes saved
+ Code + Text
[26] #Tugas 4

y_predtest = lin_reg.predict(X_test)

#MSE
MSE_train = mean_squared_error(Y_train, y_predtrain)
print('Nilai MSE data training = ', MSE_train)
MSE_test = mean_squared_error(Y_test, y_predtest)
print('Nilai MSE data testing = ', MSE_test)

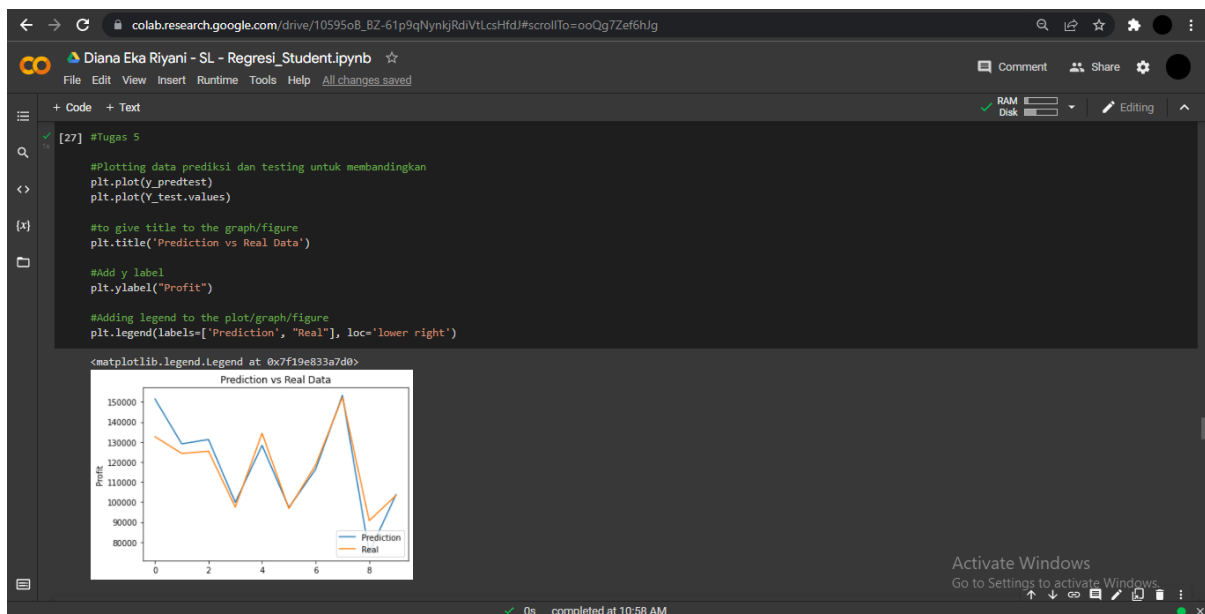
#RMSE
RMSE_train = np.sqrt(MSE_train)
print('Nilai RMSE data training = ', RMSE_train)
RMSE_test = np.sqrt(MSE_test)
print('Nilai RMSE data testing = ', RMSE_test)

#MAE
MAE_train = mean_absolute_error(Y_train, y_predtrain)
print('Nilai MAE data training = ', MAE_train)
MAE_test = mean_absolute_error(Y_test, y_predtest)
print('Nilai MAE data testing = ', MAE_test)

Nilai MSE data training = 81897487.02905282
Nilai MSE data testing = 71176114.65887138
Nilai RMSE data training = 9049.72303554384
Nilai RMSE data testing = 8436.593782971382
Nilai MAE data training = 6843.0077696397175
Nilai MAE data testing = 6843.0077696397175
```

Untuk mengetahui apakah nilai ini cukup bagus atau tidak, kita perlu membuat model regresi yang lain lalu membandingkan MSE, RMSE, dan MAE-nya. Model terbaik adalah yang MSE, RMSE, dan MAE-nya paling kecil.

Tugas 5: Visualisasi Prediksi



Terlihat bahwa nilai prediksi dan data testing juga cukup dekat.

colab.research.google.com/drive/10595oB_BZ-61p9qNynkjRdiVtLcsHfdJ#scrollTo=XJztZcxdnNxb

Diana Eka Riyani - SL - Regresi_Student.ipynb

File Edit View Insert Runtime Tools Help All changes saved

+ Code + Text

Koefisien Determinasi (R²)

```
[28] print(f'R^2 score: {lin_reg.score(X, Y)}')
```

R² score: 0.9499867945573266

Terlihat nilai R² = 0.9499, ini merupakan nilai yang sangat bagus, ini menunjukkan 94.99% dari profit dapat diprediksi oleh R&D spend dan Marketing Spend. Sisanya (5.01%) dipengaruhi faktor lain yang tidak ada di model ini

colab.research.google.com/drive/10595oB_BZ-61p9qNynkjRdiVtLcsHfdJ#scrollTo=XJztZcxdnNxb

Diana Eka Riyani - SL - Regresi_Student.ipynb

File Edit View Insert Runtime Tools Help All changes saved

+ Code + Text

Uji Simultan, Parsial, dan besar pengaruh feature

```
linreg.summary()
```

OLS Regression Results

Dep. Variable:	Profit	R-squared:	0.957
Model:	OLS	Adj. R-squared:	0.954
Method:	Least Squares	F-statistic:	408.7
Date:	Tue, 15 Mar 2022	Prob (F-statistic):	5.97e-26
Time:	03:52:32	Log Likelihood:	-421.18
No. Observations:	40	AIC:	848.4
Df Residuals:	37	BIC:	853.4
Df Models:	2		
Covariance Type:	nonrobust		
	coef	std err	t
const	4.639e+04	2887.411	16.068
x1	0.8251	0.045	18.249
x2	0.0236	0.017	1.378
Omnibus:	13.108	Durbin-Watson:	2.147
Prob(Omnibus):	0.001	Jarque-Bera (JB):	16.774
Skew:	-0.964	Prob(JB):	0.000228
Kurtosis:	5.520	Cond. No.	4.98e+05

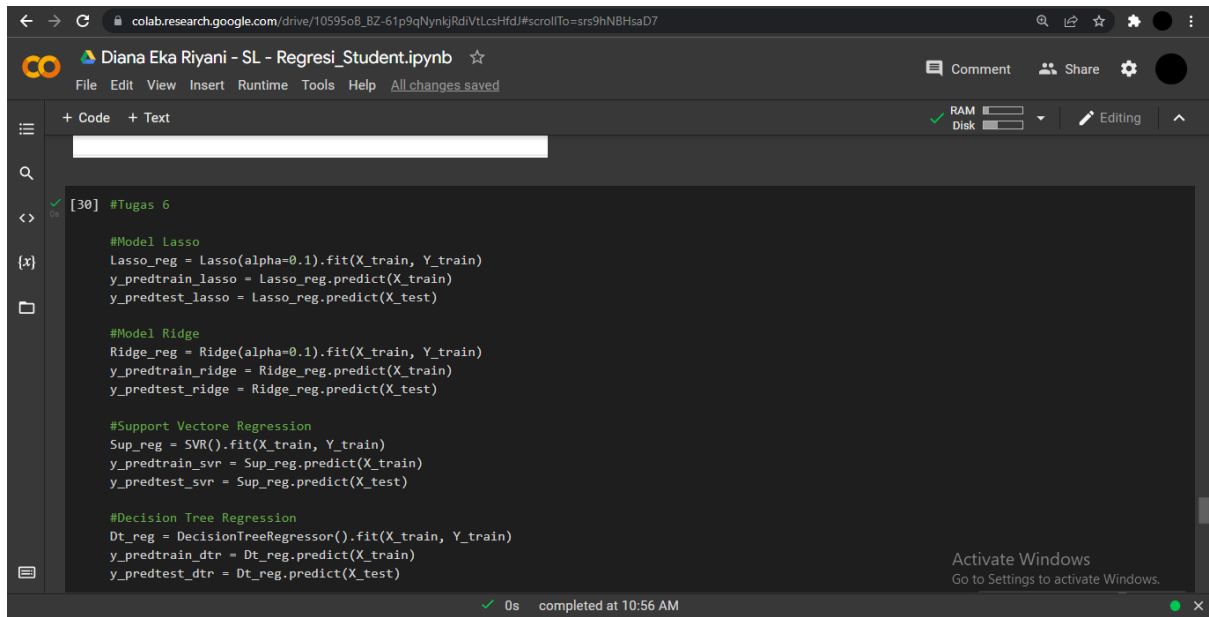
Warnings:

[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.

[2] The condition number is large, 4.98e+05. This might indicate that there are strong multicollinearity or other numerical problems.

0s completed at 10:52 AM

Tugas 6: Training 4 Model



```
[30] #Tugas 6

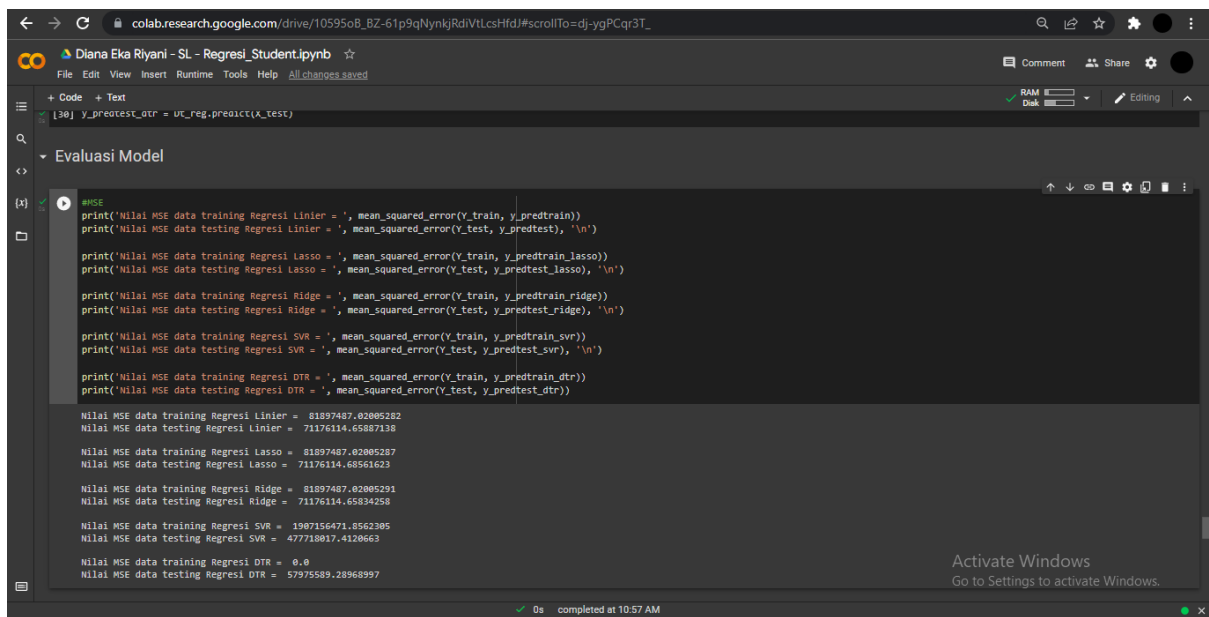
#Model Lasso
Lasso_reg = Lasso(alpha=0.1).fit(X_train, Y_train)
y_predtrain_lasso = Lasso_reg.predict(X_train)
y_predtest_lasso = Lasso_reg.predict(X_test)

#Model Ridge
Ridge_reg = Ridge(alpha=0.1).fit(X_train, Y_train)
y_predtrain_ridge = Ridge_reg.predict(X_train)
y_predtest_ridge = Ridge_reg.predict(X_test)

#Support Vector Regression
Sup_reg = SVR().fit(X_train, Y_train)
y_predtrain_svr = Sup_reg.predict(X_train)
y_predtest_svr = Sup_reg.predict(X_test)

#Decision Tree Regression
Dt_reg = DecisionTreeRegressor().fit(X_train, Y_train)
y_predtrain_dtr = Dt_reg.predict(X_train)
y_predtest_dtr = Dt_reg.predict(X_test)
```

0s completed at 10:56 AM



```
[30] y_predtest_dtr = Dt_reg.predict(X_test)

# Evaluasi Model

#MSE
print('Nilai MSE data training Regresi Linier = ', mean_squared_error(Y_train, y_predtrain))
print('Nilai MSE data testing Regresi Linier = ', mean_squared_error(Y_test, y_predtest), '\n')

print('Nilai MSE data training Regresi Lasso = ', mean_squared_error(Y_train, y_predtrain_lasso))
print('Nilai MSE data testing Regresi Lasso = ', mean_squared_error(Y_test, y_predtest_lasso), '\n')

print('Nilai MSE data training Regresi Ridge = ', mean_squared_error(Y_train, y_predtrain_ridge))
print('Nilai MSE data testing Regresi Ridge = ', mean_squared_error(Y_test, y_predtest_ridge), '\n')

print('Nilai MSE data training Regresi SVR = ', mean_squared_error(Y_train, y_predtrain_svr))
print('Nilai MSE data testing Regresi SVR = ', mean_squared_error(Y_test, y_predtest_svr), '\n')

print('Nilai MSE data training Regresi DTR = ', mean_squared_error(Y_train, y_predtrain_dtr))
print('Nilai MSE data testing Regresi DTR = ', mean_squared_error(Y_test, y_predtest_dtr))

Nilai MSE data training Regresi Linier = 81897487.02005282
Nilai MSE data testing Regresi Linier = 71176114.65887138

Nilai MSE data training Regresi Lasso = 81897487.02005287
Nilai MSE data testing Regresi Lasso = 71176114.68561623

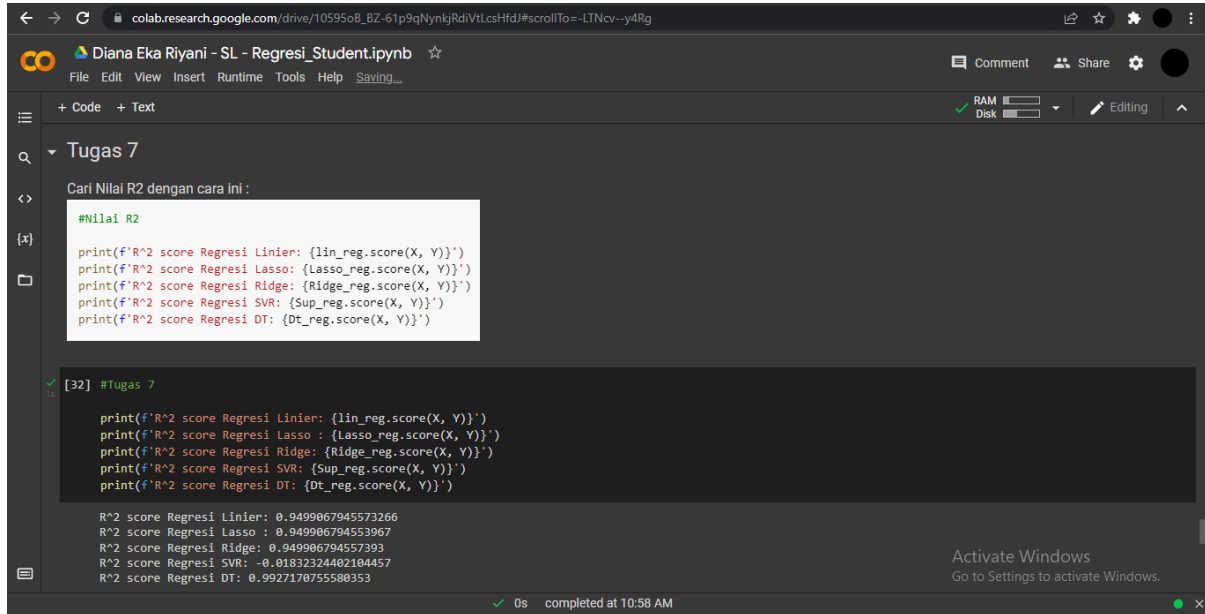
Nilai MSE data training Regresi Ridge = 81897487.02005291
Nilai MSE data testing Regresi Ridge = 71176114.65834258

Nilai MSE data training Regresi SVR = 1987156471.8562385
Nilai MSE data testing Regresi SVR = 477718817.4120663

Nilai MSE data training Regresi DTR = 0.0
Nilai MSE data testing Regresi DTR = 57975589.28968997
```

0s completed at 10:57 AM

Tugas 7: Mencari Nilai R2



```
#Nilai R2

print(f'R^2 score Regresi Linier: {lin_reg.score(X, Y)}')
print(f'R^2 score Regresi Lasso: {Lasso_reg.score(X, Y)}')
print(f'R^2 score Regresi Ridge: {Ridge_reg.score(X, Y)}')
print(f'R^2 score Regresi SVR: {Sup_reg.score(X, Y)}')
print(f'R^2 score Regresi DT: {Dt_reg.score(X, Y)}')
```

```
[32] #Tugas 7

print(f'R^2 score Regresi Linier: {lin_reg.score(X, Y)}')
print(f'R^2 score Regresi Lasso : {Lasso_reg.score(X, Y)}')
print(f'R^2 score Regresi Ridge: {Ridge_reg.score(X, Y)}')
print(f'R^2 score Regresi SVR: {Sup_reg.score(X, Y)}')
print(f'R^2 score Regresi DT: {Dt_reg.score(X, Y)}')
```

```
R^2 score Regresi Linier: 0.9499867945573266
R^2 score Regresi Lasso : 0.949986794553967
R^2 score Regresi Ridge: 0.949986794557393
R^2 score Regresi SVR: -0.01832324402104457
R^2 score Regresi DT: 0.9927170755580353
```

- Model regresi Support Vector Regression memiliki MSE yang terlalu tinggi dibandingkan model regresi lainnya dan R2 yang sangat rendah, sehingga kemungkinan pada model ini terjadi underfitting.
- Model regresi Decision Tree Regression R2 paling bagus, mMSE data training 0 (sangat kecil), sementara MSE data testing jauh di atasnya, ini menunjukkan pada model ini terjadi overfitting.
- Model regresi linier, regresi Lasso, dan regresi Ridge memiliki MSE yang tidak jauh antara testing dan trainingnya. Nilai R2nya pun sangat bagus, hampir 95%. Ini menunjukkan model ini sudah good fit.