**Klasifikasi Jenis Kanker Payudara Menggunakan**

**Pendekatan Model pada *Machine Learning***

**Makalah Ini Disusun Sebagai Proyek UAS Kelompok Sains Data**

**Fasilitator: Bevina Desjwiandra Handari**



**Disusun oleh:**

**Deanarani Kharisma (2106726125)**

**Dimas Satrio Adjie (2106651276)**

**Roswita Bellinda (2106722505)**

**Salsabila H. Sastro (2106722726)**

**FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM**

**UNIVERSITAS INDONESIA**

**DEPOK**

**2023**

**Abstrak**

Makalah ini akan membahas salah satu aplikasi *Neural* *Network* yang digunakan untuk mengklasifikasikan jenis kanker payudara. *Artificial* *Neural* *Network* adalah jaringan dari sekelompok unit pemroses kecil (*neural*/neuron) yang dimodelkan dan ditiru berdasarkan seperti jaringan saraf manusia, yaitu neuron. *Artificial* *Neural* *Network* merupakan suatu sistem yang dapat beradaptasi dengan mengubah struktur-strukturnya dan akan menyelesaikan suatu masalah berdasarkan informasi-informasi yang mengalir baik itu eksternal maupun internal.

Kata kunci: Data, Klasifikasi, *Machine* *Learning*, *Neural* *Network*, *Artificial Neural Network*.

1. **PENDAHULUAN**
   1. **Latar Belakang**

Kanker payudara adalah kanker yang paling sering terjadi pada wanita, berdampak pada 2,1 juta wanita setiap tahun, dan juga menyebabkan kematian terkait kanker pada wanita. Diperkirakan 627.000 wanita meninggal karena kanker payudara, yaitu sekitar 15% dari semua kematian akibat kanker di kalangan wanita.

​​Sel kanker payudara yang pertama dapat tumbuh menjadi tumor sebesar 1 cm pada waktu 8–12 tahun. Sel kanker tersebut diam pada kelenjar payudara. Sel-sel kanker payudara ini dapat menyebar melalui aliran darah ke seluruh tubuh. Sel kanker payudara dapat bersembunyi di dalam tubuh kita selama bertahun-tahun tanpa kita ketahui, dan tiba-tiba aktif menjadi tumor ganas atau kanker.

Secara umum, terdapat dua jenis tumor payudara, yaitu tumor payudara ganas dan tumor payudara jinak. Keduanya tentu tidak sama, perbedaan paling mendasar antara tumor payudara ganas dan jinak adalah perkembangannya. Secara umum, tumor ganas berkembang menjadi sel kanker, sedangkan tumor jinak adalah non-kanker dan dapat sembuh dengan sendirinya, tetapi mungkin memerlukan perawatan khusus.

Berdasarkan uraian di atas, dalam pemodelan masalah ini, kami berencana untuk membuat model *neural* *network* sebagai alternatif penyelesaian masalah klasifikasi kanker payudara. Model *neural* *network* yang dibuat memanfaatkan data dari 32 fitur, terdiri atas *radius*, *texture*, *perimeter*, *area*, *smoothness*, *compactness*, dan fitur-fitur lainnya yang akan disebutkan rinci di bagian selanjutnya. Diagnosis nantinya akan dipilih sebagai variabel output dan komponen lainnya sebagai variabel prediktor. Pada akhir pemodelan ini, besar harapan kami akan memperoleh model *neural* *network* yang dapat mengestimasi dan mengklasifikasikan jenis kanker payudara pada keadaan tertentu, yang mana modelnya sendiri juga harus memenuhi standar kelayakan model yang ada.

* 1. **Rumusan Masalah**

Rumusan masalah dalam penelitian kami sebagai berikut.

1. Bagaimana akurasi klasifikasi model *neural* *network* berdasarkan data yang tersedia?
2. Bagaimana hubungan antara diagnosis dengan fitur-fitur lainnya yang digunakan dalam pemodelan?
3. Bagaimana hasil analisis data terhadap model *neural* *network* yang diperoleh?
   1. **Tujuan Penelitian**

Berdasarkan rumusan masalah di atas, tujuan penelitian kami adalah sebagai berikut.

1. Mengetahui akurasi klasifikasi model *neural* *network* berdasarkan data yang tersedia.
2. Mengetahui hubungan antara diagnosis dengan fitur-fitur lainnya yang digunakan dalam pemodelan.
3. Mengetahui hasil analisis data terhadap model *neural* *network* yang diperoleh.
4. **ISI**
5. **Data dan Metode**

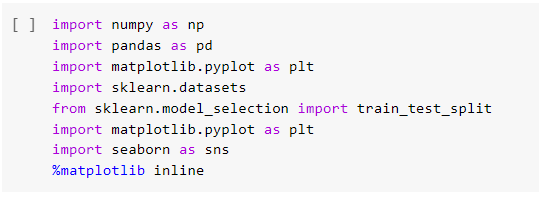
Model *neural* *network* untuk masalah klasifikasi jenis kanker payudara yang dibuat memanfaatkan *dataset* Breast Cancer Wisconsin (sumber: [https://archive.ics.uci.edu/](https://archive.ics.uci.edu/ml/datasets/breast+cancer+wisconsin+(diagnostic))) yang terdiri dari 569 baris dan 32 fitur. Namun, yang digunakan hanya menjadi 28 fitur kontinu yang memiliki korelasi yang paling baik dengan variabel target. Variabel target yang dimaksud jenis tumor kanker payudaranya, apakah tumor ganas (malignant) atau tumor jinak (benign). Inilah yang nantinya akan menjadi output atau tujuan dari model *neural* *network* yang kami bangun. Berikut fitur-fitur beserta penjelasannya pada *dataset* kami:

1. ID: Nomor pasien
2. Diagnosis: Apakah tumornya ganas atau jinak
3. Radius: Jarak titik tengah dengan lingkaran payudaranya
4. Texture: Standar deviasi dari nilai skala abu-abu
5. Perimeter: Besarnya tumor
6. Area: Daerah terkenanya tumor
7. Smoothness: Variasi lokal dengan panjang radius
8. Compactness = (perimeter^2) / (area - 1.0)
9. Concavity: Kerasnya bagian cekung dari garis bentuknya
10. Concave points: Banyaknya bagian cekung dari garis bentuknya
11. Symmetry: Kesimetrisan
12. Fractal dimension = aproksimasi coastline - 1
13. Mean: Rata-rata
14. Se: Standard error
15. Worst: Rata-rata terbesar

Adapun metode yang kami gunakan dalam pemodelan ini adalah pengolahan, analisis, dan visualisasi data numerik dengan menggunakan Google Colaboratory dengan bahasa pemrograman Python pada tautan berikut [project-sainsdata-kelompok1](https://colab.research.google.com/drive/1NkM9WpC1abbAAcMnaFIebxqLvtENBhLd?usp=sharing). Proses pemodelan secara garis besar terdiri atas *import* data, *data cleaning* dan *data preprocessing*, membangun model dengan Keras dan TensorFlow, *train* dan melihat akurasi model*,* dan terakhir *test* model dengan sebuah contoh input.

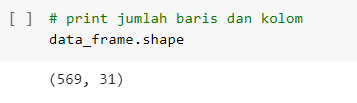
1. **Implementasi**

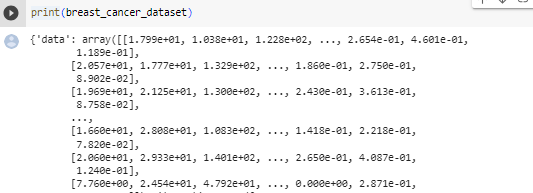
**2.2.1 *Importing the Dependencies***

Mengimpor beberapa fungsi dan *libraries* seperti *numpy, pandas, matplotlib.pyplot, sklearn datasets,* dan *train test* untuk bisa menjalankan program ini.

**2.2.2 *Data Importation***

* Dalam penelitian ini, kami menggunakan data sebanyak 569 baris dengan 31 fitur, yang semuanya adalah data numerik.



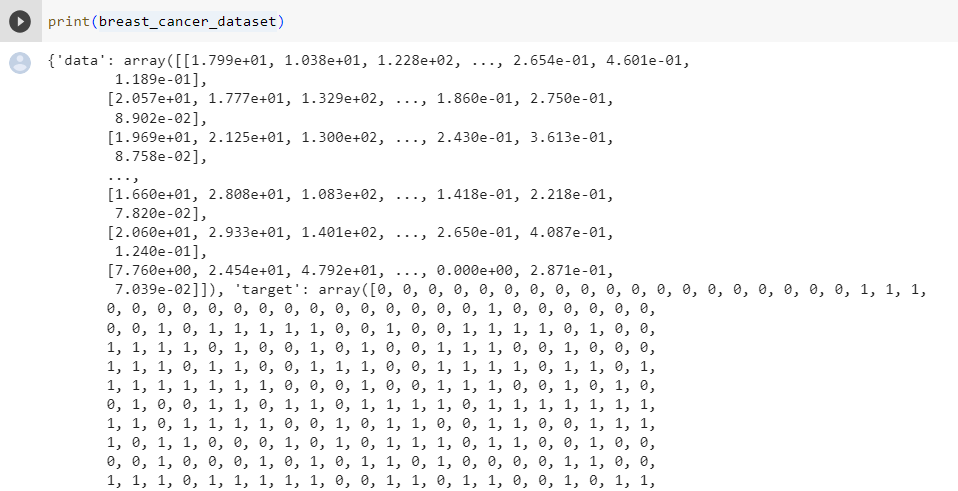


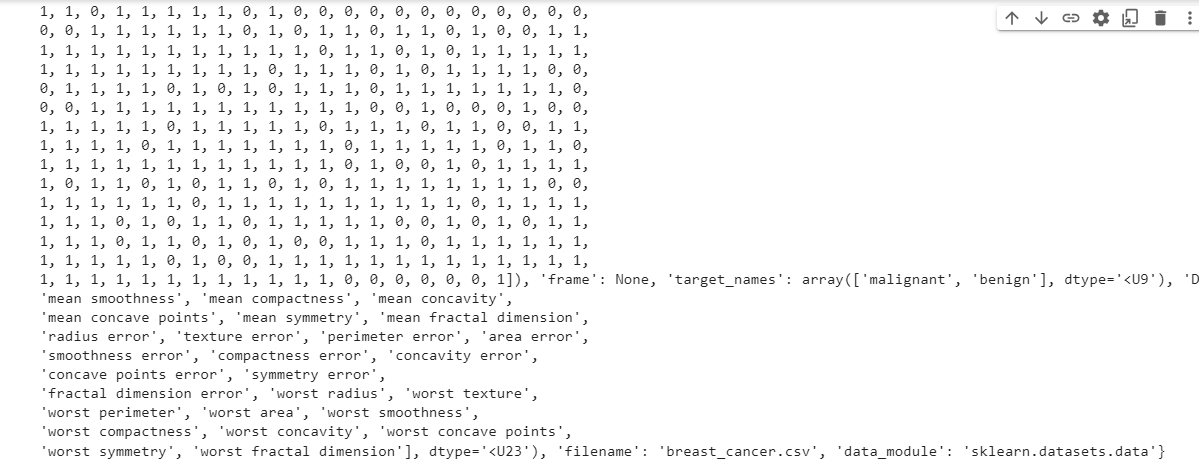
**2.2.3 *Data Collection & Processing***

* Menggunakan fungsi berikut untuk memanggil *dataset* kanker payudara.



* Berikut hasilnya dapat dilihat menampilkan nilai fitur dan elemen. Serta target dimana 0 mewakili tumor ganas dan 1 mewakili tumor jinak.





**2.2.4 Membuat Data Frame**

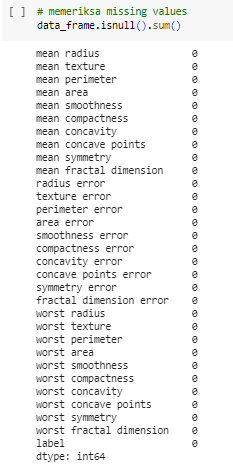
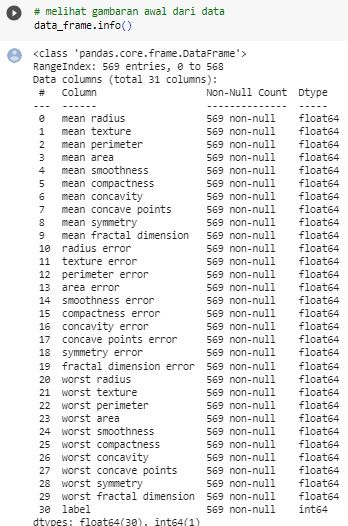
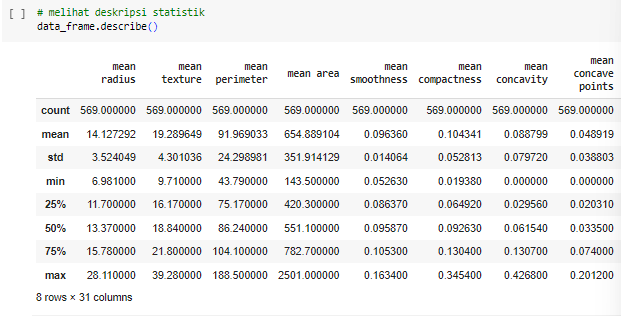
* Menggunakan pandas library untuk membuat data frame yang berisi kolom berupa fitur-fitur dari *dataset*.



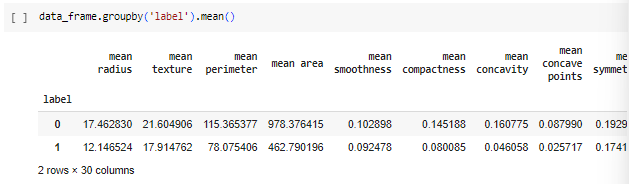
* Membuat kolom baru untuk fitur target yang diberi nama “label”.



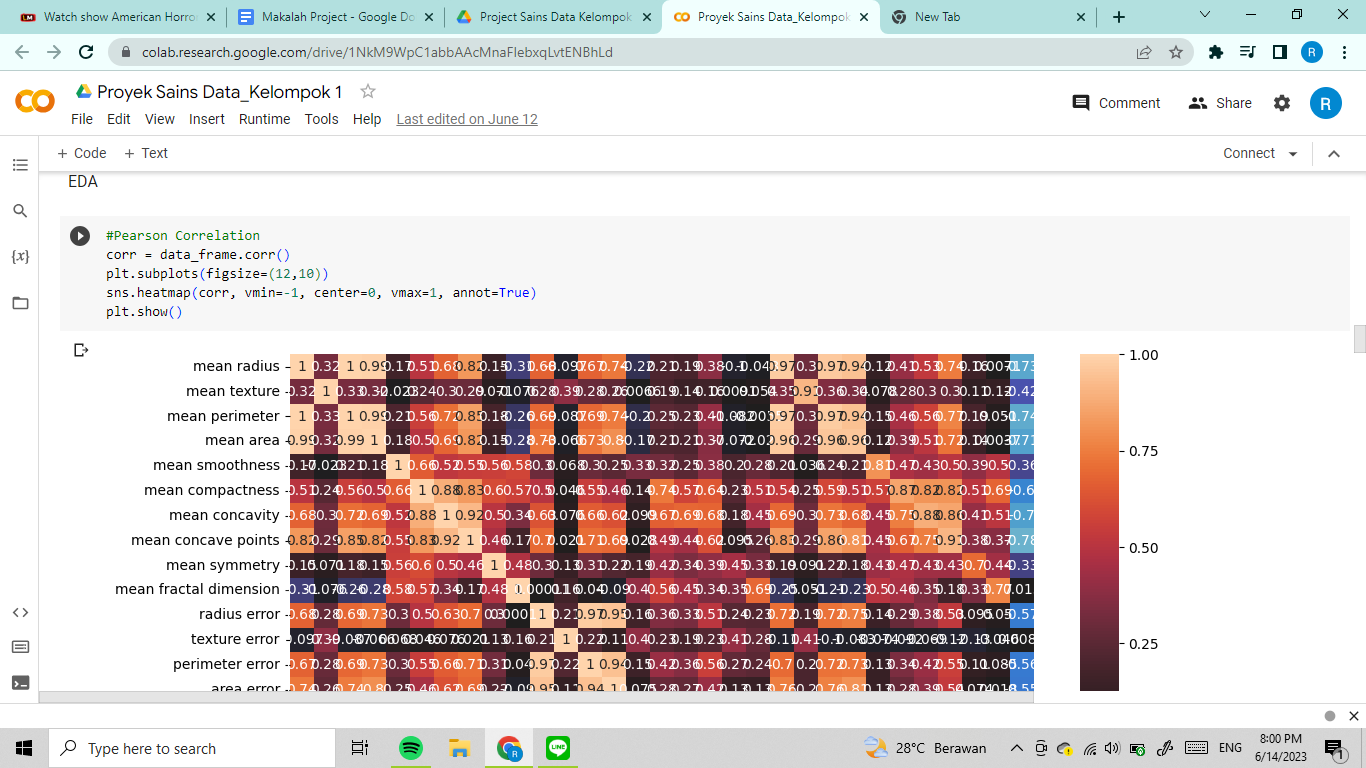
* Mencari informasi data menggunakan fungsi berikut untuk melihat berapa banyak null dan non-null pada data, sehingga dapat dibersihkan, serta statistik data.

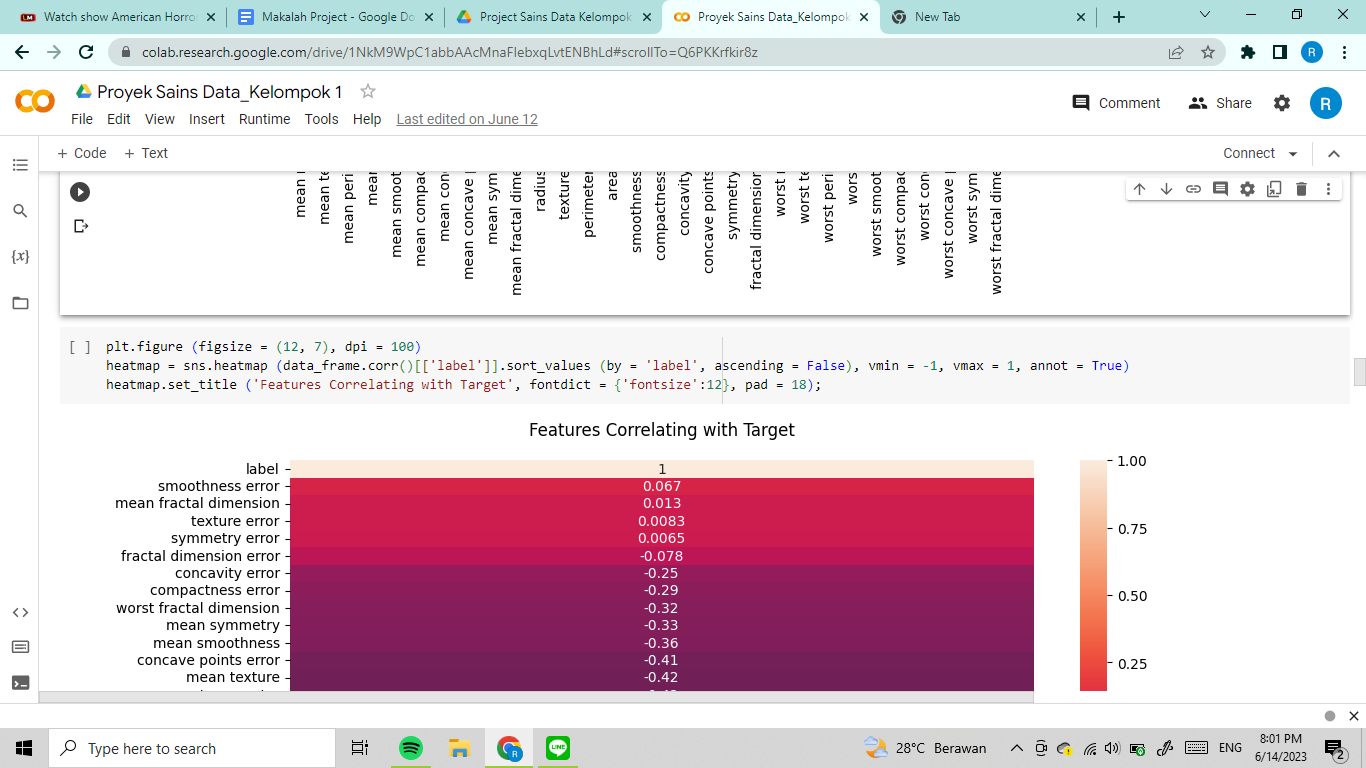


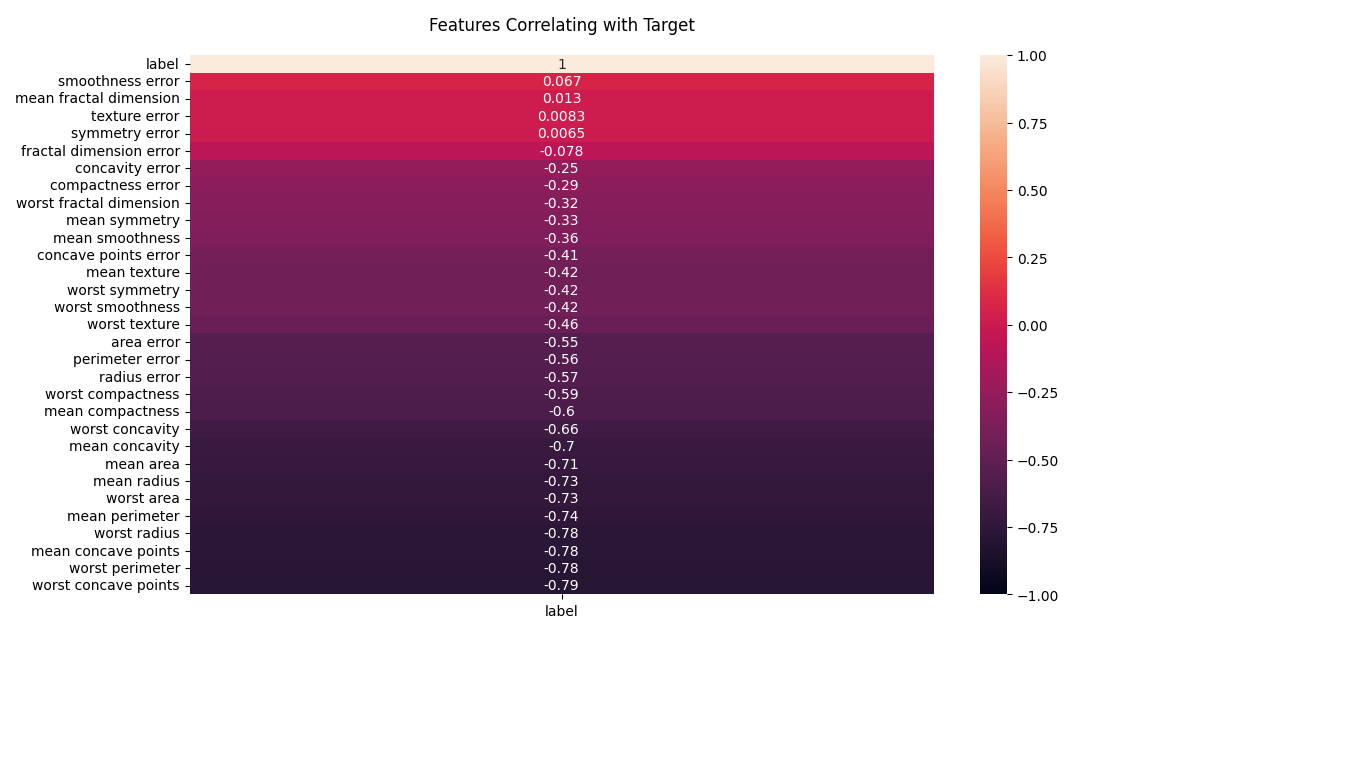
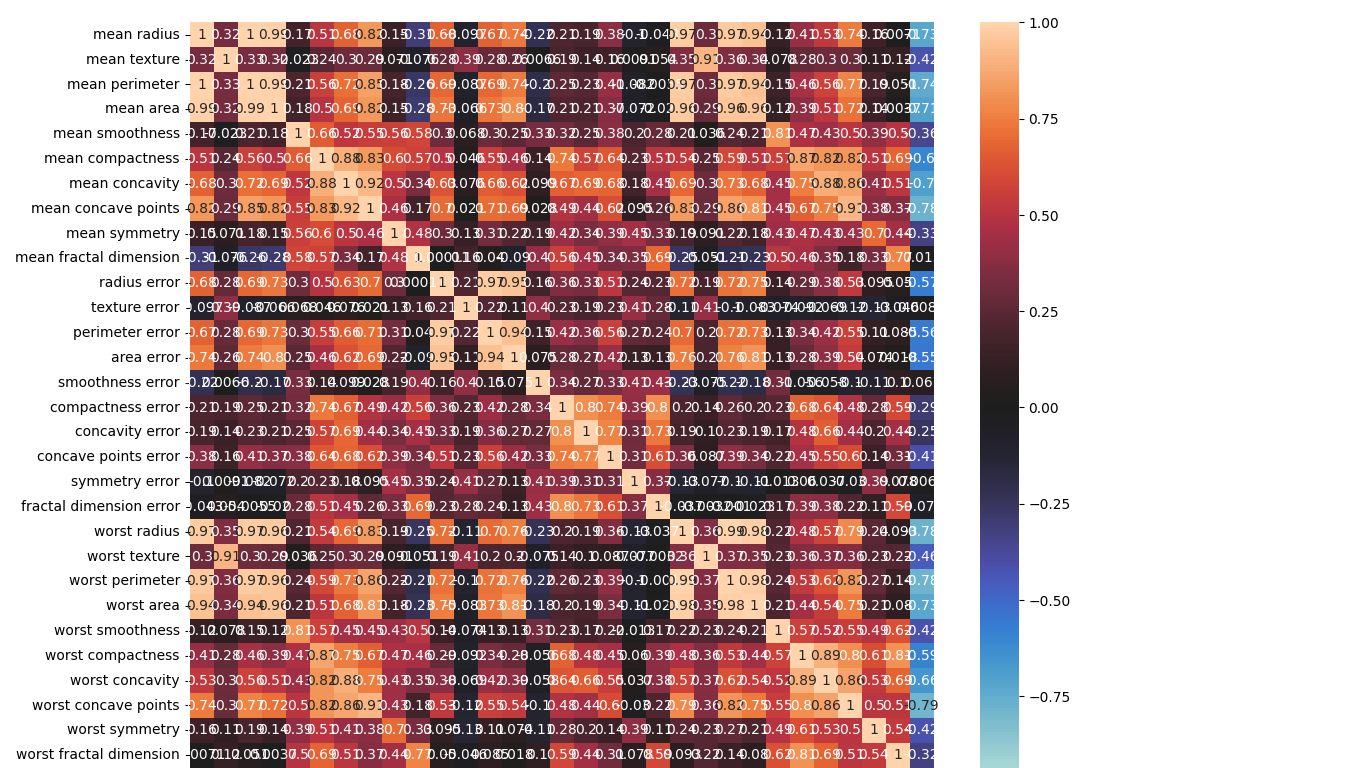
* Melakukan pengelompokkan data dan mencari rata-ratanya untuk setiap label.



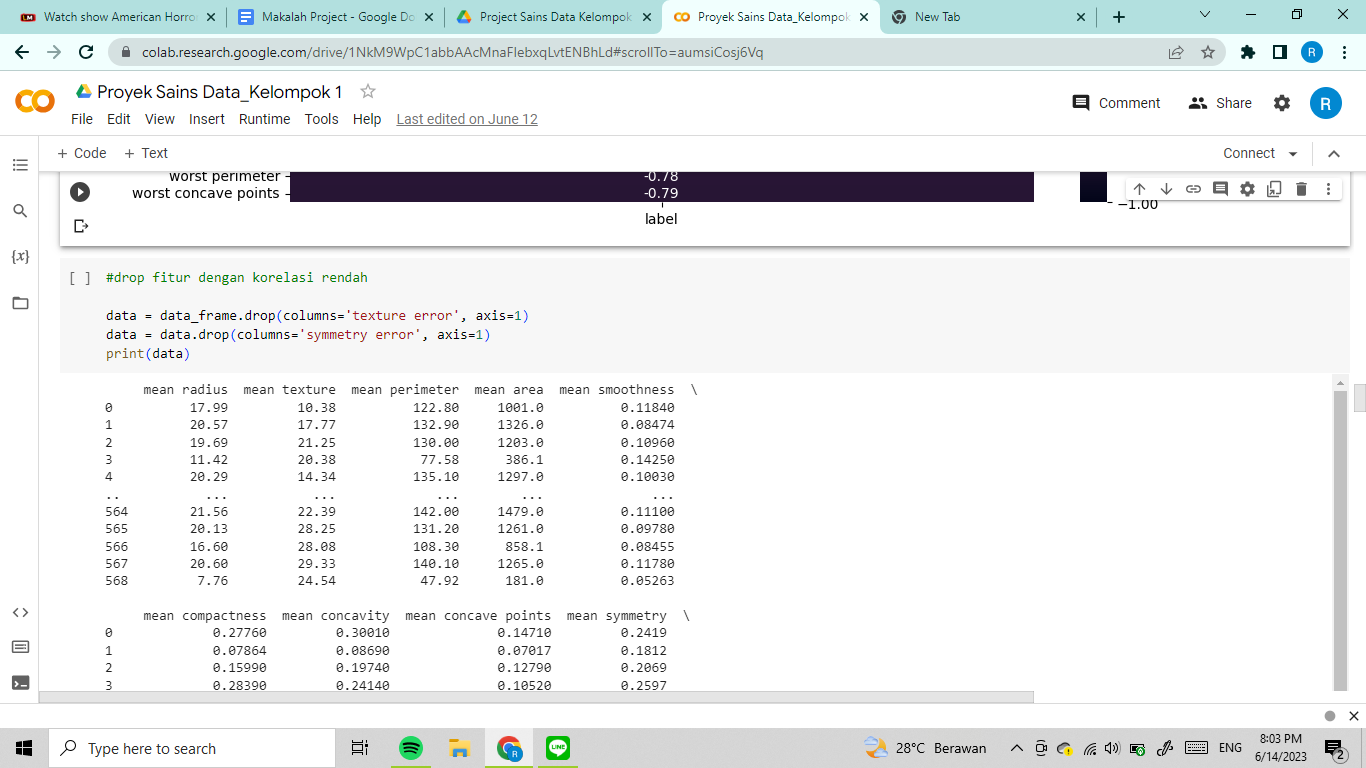
**2.2.5 *Exploratory Data Analysis***

* Mencari *Pearson Correlation* antarfitur dan antara fitur serta target.





* *Drop* fitur dengan korelasi rendah, yaitu “*texture* *error*” dan “*symmetry* *error*”.



**2.2.6 Memisahkan fitur dengan target**

Dalam *machine learning* ataupun *neural* *network,* fitur dan target haruslah dipisah, oleh karena itu kami melakukan beberapa hal berikut.

* Membuat variabel baru, yaitu X dan Y



Dimana X berisi semua kolom kecuali label dan Y berisi kolom label.

**2.2.7 Membagi data menjadi *data test* dan *data train***

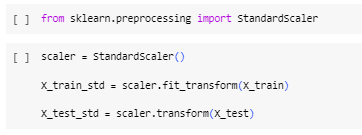
Sebelum membuat *neural* *network*, pastikan mengambil beberapa persen dari data untuk diuji.



* X\_train berisi data pelatihan dari fitur dan label yang sesuai akan disimpan di Y\_train.
* X\_test berisi fitur data uji dan semua label yang akan disimpan di Y\_test.
* Dengan menggunakan *test\_size* = 0,2 karena kami menggunakan 20% data menjadi data uji dengan *random\_state* = 2.

**2.2.8 Standarisasi Data**

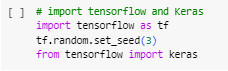
Untuk meningkatkan tingkat akurasi kebenaran data, sangat penting untuk melakukan standarisasi data.



**2.2.9 *Deep Neural Network***

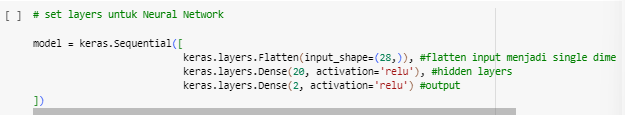
Bagian terpenting adalah membangun *neural network*, berikut langkah yang kami gunakan.

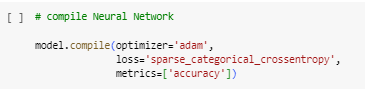
* Import tensorflow dan keras.



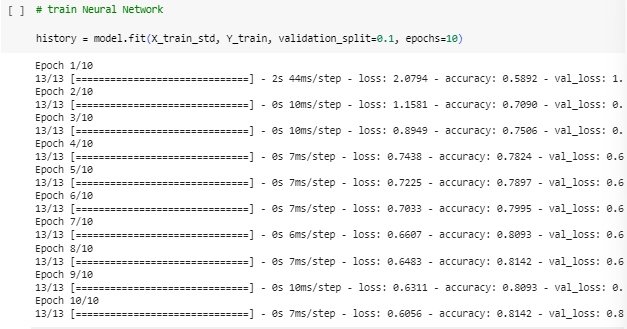
Tensorflow adalah perpustakaan pembelajaran mendalam yang dibuat oleh Google sehingga memudahkan kita untuk membuat *neural network,* sedangkan Keras adalah pembungkus tensorflow. Random seed digunakan untuk mendapatkan nilai akurasi yang sama berapa kali kita menjalankan codenya.

* Membuat layer dan *neural* *network*.



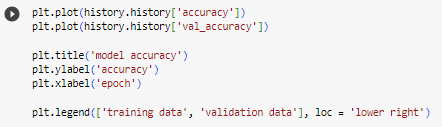


* Melakukan pelatihan *neural* *network*.

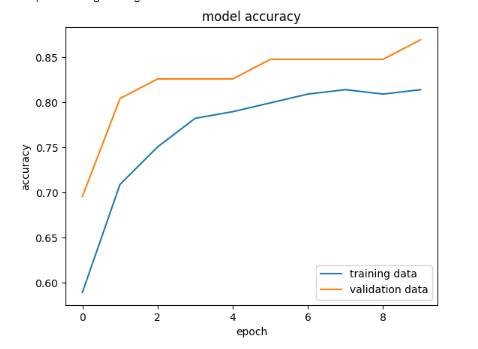
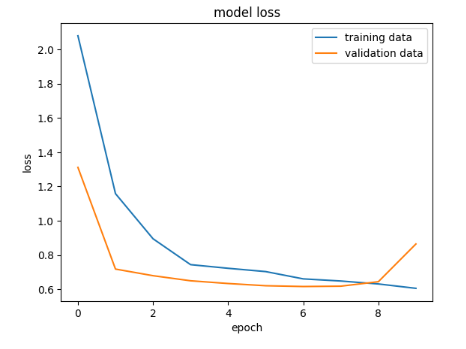


**2.2.10 *Visualizing Accuracy and Loss***

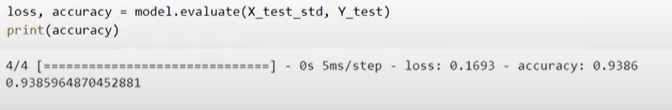
Untuk melihat tingkat keakuratan model dan kesalahan maka dibuat visualisasi dari variabel ‘history’ yang berisi *training data* agar mudah dimengerti menggunakan fungsi berikut.



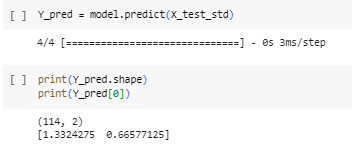
Lalu menghasilkan grafik akurasi model dari *train data* dan *validation data* serta tingkat *loss* sebagai berikut, dengan *validation data* adalah data yang sebagian besar digunakan untuk mendeskripsikan evaluasi dari model saat *hyperparameter tuning* dan *data prepration*, sedangkan *test data* adalah data yang digunakan untuk mendeskripsikan evaluasi dari model final untuk dibandingkan dengan model final lainnya.

**2.2.11 Mengecek keakuratan mode pada *data* *test***

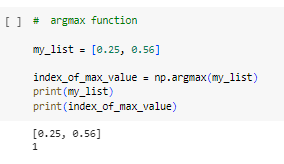
* Bisa dilihat bahwa tingkat akurasi model adalah 93,85%



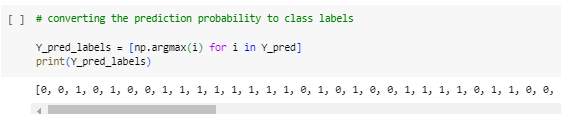
* Membangun model.predict() untuk memberikan nilai probabilitas kelas untuk *data point*.



* Memanggil fungsi *argmax* untuk pengklasifikasian.



* Melakukan tes ulang untuk model.predict() dan fungsi *argmax* untuk memprediksi probabilitas label.



**2.2.12 *Building the predictive system***

Tahap terakhir adalah membangun sistem prediksi, berikut algoritma yang kami gunakan.

* Memasukkan data dari 30 fitur yang dipakai.



* Mengubah data yang sudah diinput menjadi *numpy array*.

**

* Membentuk kembali *numpy array* seperti yang kami perkirakan untuk satu *data point*.
* Melakukan standarisasi data agar tingkat akurasi kebenaran tinggi.



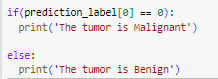
* Membuat model.prediction() dari data yang sudah diinput.



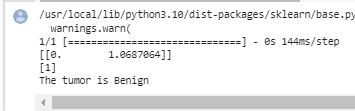
* Memasukkan fungsi *argmax* agar dapat diklasifikasikan ke label atau target.



* Mencetak hasil, apabila *prediction\_label* sama dengan 0 maka tumor tersebut ganas, selain nilai 0 maka tumor tersebut jinak.



* Berikut contoh hasil yang ditunjukkan dengan menggunakan salah satu data pada data set.



1. **Analisis Data**

Inferensi dari EDA (*Exploratory Data Analysis*) dan *Data Test*:

1. Tidak ada nilai yang hilang.
2. Semua nilainya adalah nilai numerik kontinu.
3. Mean sedikit lebih tinggi daripada median untuk sebagian besar fitur. Oleh karena itu, data tersebut condong ke kanan (*right skewed*).
4. Terdapat sedikit ketidakseimbangan *dataset*, yaitu kasus Benign (yang diwakilkan dengan 1) lebih banyak daripada kasus Malignant (yang diwakilkan dengan 0).
5. Mean sebagian besar fitur jelas lebih besar untuk kasus Malignant dibandingkan dengan kasus Benign (*Groupby*).
6. Sebagian besar fitur memiliki pencilan (*outliers*).
7. Matriks korelasi mengungkapkan bahwa sebagian besar fitur memiliki korelasi yang tinggi. Oleh karena itu, kita dapat menghapus beberapa fitur tertentu selama Seleksi Fitur.
8. **PENUTUP**
9. **Kesimpulan**
10. Penerapan model *Artificial* *Neural* *Network* yang digunakan untuk mengklasifikasikan jenis kanker payudara memberikan hasil yang relatif baik pada proses pengolahan data dengan tingkat akurasi model 93,85%.
11. Model *neural* *network* yang digunakan dengan memanfaatkan *dataset* Breast Cancer Wisconsin dalam masalah klasifikasi jenis kanker payudara terdiri dari 569 baris dan 32 fitur. Namun, hanya 28 fitur kontinu yang memiliki korelasi yang paling baik dengan variabel target.
12. **Saran**

Untuk mendapatkan hasil yang lebih akurat, perlu diterapkannya *hyperparameter* *tuning* (proses mencari nilai optimal dari *hyperparameter* suatu model *machine* *learning* untuk memperbaiki performa model *machine learnin*g), yaitu suatu nilai atau parameter yang diatur secara manual dan berperan di dalam optimasi kinerja suatu model.

1. **DAFTAR PUSTAKA**

[1] Haykin, Simon, Neural Networks and Learning Machines, 4th ed. Canada: Pearson, 2016.

[2] Chollet, François, Deep Learning with Python, 2nd ed. Manning, 2021.

[3] Nielse, Michael A., Neural Networks and Deep Learning. Determination Press, 2015.