1. Jelaskan cara kerja dari algoritma tersebut! (boleh dalam bentuk *pseudocode* ataupunnarasi)
2. Bandingkan hasil evaluasi model pada nomor 2 dan 3, bagaimana hasil perbandingannya? Jika ada perbedaan, jelaskan alasannya!
3. Jelaskan *improvement* apa saja yang bisa Anda lakukan untuk mencapai hasil yang lebih baik dibandingkan dengan hasil yang Anda punya saat ini! *Improvement* yang dimaksud tidak terbatas pada bagaimana algoritma diimplementasikan, namun juga mencakup tahap sebelum *modeling and validation*.
4. **KNN regression**
5. Cara kerja Algoritma

KNN bekerja dengan membandingkan titik data baru dengan titik data yang sudah ada dalam dataset berdasarkan jarak (misalnya Euclidean). K titik data terdekat dinamakan titik tetangga/neighbor, dengan k adalah hyperparameter. Titik data baru akan diberi nilai berdasarkan k tetangga terdekatnya, biasanya rata-rata nilai dari titik tetangganya.

1. Perbandingan Hasil Evaluasi Model

Prediksi yang dihasilkan sama persis. Model saya melakukan kalkulasi yang jauh lebih lama daripada model dari scikitlearn. Hal ini terjadi salah satunya karena model dari scikit learn menggunakan sparse matrix untuk mempercepat kalkulasi dan mengurangi memori.

1. Improvement

Menggunakan sparse matrix untuk mempercepat kalkulasi

1. **Polynomial Regression**
2. Cara kerja Algoritma

Polynomial Regression memiliki cara kerja yang mirip dengan linear regression. Algoritma ini akan meluaskan (*expand*) titik-titik masukan sesuai dengan degree yang ditetapkan. Misal jika inputnya X dengan degree n, maka akan diekspan menjadi Xn, Xn-1…, X. Bisa juga ditambahkan cross variable jika fitur input lebih dari satu.

Setelah diekspan, akan dibuat variable weight pada tiap-tiap variabel ekspan. Variabel weight akan diubah terus untuk meminimalkan loss berdasarkan loss function yang digunakan (biasanya MSE) dengan menggunakan optimizer (contoh: gradient descent).

1. Perbandingan Hasil Evaluasi Model

Saya menggunakan scikit learn linear regression yang input datanya sudah di-expand dengan polynomial features. Score yang dihasilkan oleh model dari scikit learn sedikit lebih baik daripada model yang saya buat. Waktu kalkulasi model pada scikit learn juga sedikit lebih cepat dari model saya. Saya menggunakan learning rate senilai 0.02, iterasi sejumlah 10000, dan degree polynomial bernilai 3. Perbedaan nilai score terjadi karena hyperparameter linear regression pada scikit learn tidak sama dengan hyperparameter pada model saya (sepertinya scikit learn menggunakan metode rumus matriks, bukan SGD). Jika iterasi saya tambah, saya yakin nilai scorenya bisa menyamai score yang dihasilkan oleh model dari scikit learn.

1. Improvement

Menggunakan parallel programming untuk mempercepat kalkulasi.

1. **Regresion Tree**
2. Cara kerja Algoritma

Secara garis besar, Regression Tree menggunakan struktur Tree yang memiliki cabang dan daun untuk membagi-bagi data. Regression Tree membagi data berdasarkan threshold yang ditentukan. Jika kurang dari threshold, data masuk bagian cabang kiri, jika lebih maka data masuk bagian cabang kanan. Pembagian terus dilakukan hingga semua data termasuk dalam sebuah kumpulan data yang berupa daun.

Regression Tree memiliki beberapa variasi dalam penentuan threshold, saya menggunakan cara memilih threshold dari data unik pada dataset lalu dicari threshold yang menghasilkan nilai loss paling kecil.

1. Perbandingan Hasil Evaluasi Model

Model yang saya buat memiliki score yang sedikit lebih jelek dan waktu yang jauh lebih lama daripada model oleh scikit learn. Perbedaan waktu kalkulasi sebagian besar disebabkan oleh penggunaan dense array pada model saya sehingga ketika data masukan memiliki banyak fitur (one hot encoding dengan banyak kategori misalnya) proses kalkulasi akan memakan waktu jauh lebih banyak. Sedangkan model dari scikit learn menggunakan sparse matriks yang menghemat memori ketika matriks banyak memiliki angka nol (one hot encoding misalnya) sehingga kalkulasi diselesaikan jauh lebih cepat.

Perbedaan score disebabkan oleh perbedaan hyperparameter lanjutan, mengingat bahwa regression tree bisa memiliki banyak sekali hyperparameter yang tidak diterapkan di model saya.

1. Improvement

Untuk mempercepat waktu kalkulasi, model perlu menggunakan sparse matriks. Selain itu, penambahan hyperparameter yang lebih bervariasi juga bisa menambah fleksibilitas model dan mempermudah penggunaanya.

1. **Support Vector Regressor**
2. Cara kerja Algoritma

Secara garis besar, SVR adalah metode regresi yang didasarkan pada Support Vector Machine (SVM) untuk menemukan hyperplane yang memprediksi nilai output dalam batas margin tertentu.

1. Perbandingan Hasil Evaluasi Model

Model yang saya buat memiliki score yang jauh lebih baik daripada model dari scikit learn dengan hyperparameter yang sama. Hal ini terjadi karena scikit learn tidak menggunakan optimizer gradient descent melainkan menyelesaikanya dengan quadratic programming solver.

1. Improvement

Untuk mempercepat waktu kalkulasi, model perlu menggunakan sparse matriks. Selain itu, penambahan kernel-kernel lain juga bisa menambah versatilitas model.

1. **Artificial Neural Network (ANN)**
2. Cara kerja Algoritma

Secara garis besar, SVR adalah metode regresi yang didasarkan pada Support Vector Machine (SVM) untuk menemukan hyperplane yang memprediksi nilai output dalam batas margin tertentu.

1. Perbandingan Hasil Evaluasi Model

Model yang saya buat memiliki score yang jauh lebih baik daripada model dari scikit learn dengan hyperparameter yang sama. Hal ini terjadi karena scikit learn tidak menggunakan optimizer gradient descent melainkan menyelesaikanya dengan quadratic programming solver.

1. Improvement

Untuk mempercepat waktu kalkulasi, model perlu menggunakan sparse matriks. Selain itu, penambahan kernel-kernel lain juga bisa menambah versatilitas model.