

BAB IV

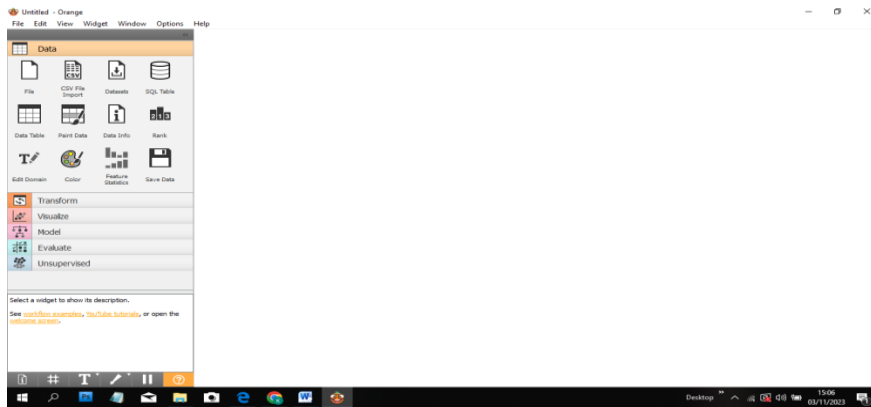
IMPLEMENTASI SISTEM

4.1 Implementasi *Orange*

Orange merupakan sebuah aplikasi pengolah data secara visual tanpa perlu membuat program. Untuk melakukan proses implementasi dalam *data mining* akan menggunakan sistem *widget* (elemen yang menampilkan informasi tertentu) dan setiap *widget* mempunyai fungsi masing-masing dan dapat menerima data *input* dan mengeluarkan *output*. Perangkat lunak *open source* untuk melakukan proses *data mining* atau analitik melalui konsep *visual programming*. Proses yang menggunakan teknik statistika, matematika, kecerdasan buatan, *machine learning* untuk mengekstraksi dan mengidentifikasi informasi dalam data yang bermanfaat dan pengetahuan yang terkait dari berbagai basis data. Setiap pola dalam data yang berjumlah besar akan diekstraksi sehingga akan mempermudah ketika anda akan melakukan pencarian atau pengolahan data lanjutan. *Orange* unggul dalam hal visualisasi atau *visual programming*. *Orange* juga menyediakan banyak *widget* yang diletakkan pada *canvas/drawing board* kemudian dihubungkan dengan *widget-widget* lainnya. Sehingga dapat mempermudah pemakaian dengan data dan melakukan proses data analitik secara intuitif. Berikut merupakan implementasi dari program yang akan diproses datanya.

4.2 Implementasi Sistem

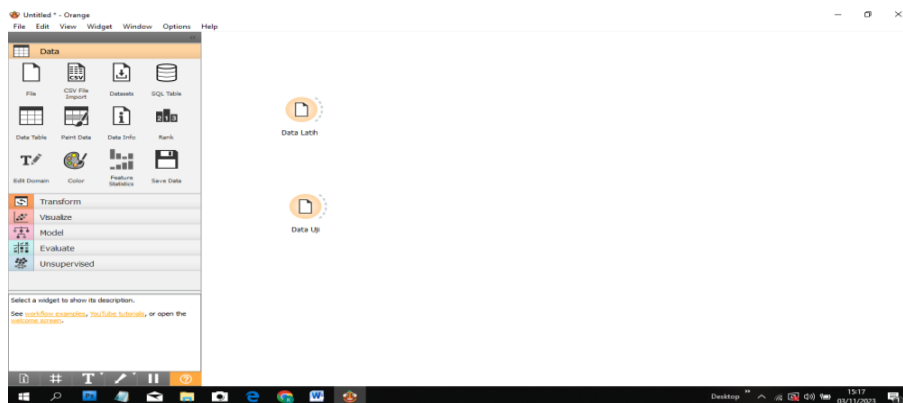
Implementasi prediksi hasil pertanian tanaman pangan menggunakan *orange*.



Gambar 4.1 Tampilan aplikasi *orange*

4.3 Tampilan *Widget File*

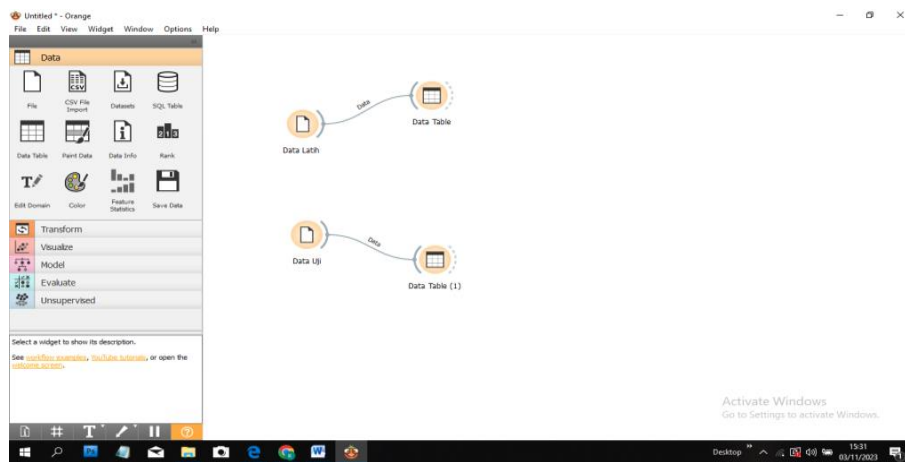
File data input dapat mengirimkan kumpulan data ke saluran *output*. *Widget* ini juga menyertakan direktori dengan kumpulan data sampel yang sudah diinstal sebelumnya dengan *orange*. *Orange* dapat meng-*input file* data *URL* dan tampilan antarmuka menu *file*, untuk *file* ini terdapat 2 submenu yaitu, data latihan dan data uji.



Gambar 4.2 *Widget* data latihan dan data uji

4.4 Tampilan *Widget Data Tabel*

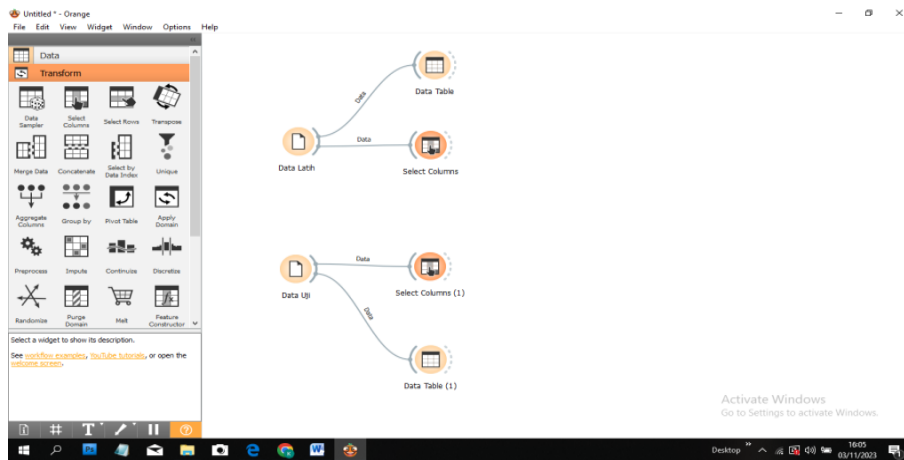
Dengan menampilkan *widget* data tabel menerima satu atau lebih kumpulan data dalam *input*-nya dan menampilkannya sebagai lembar kerja (*spreadsheet*). Data dapat diurutkan berdasarkan nilai atribut. *Widget* ini juga mendukung pemilihan *instance* data secara manual.



Gambar 4.3 Tampilan *widget* data tabel

4.5 Tampilan *Widget Select Columns*

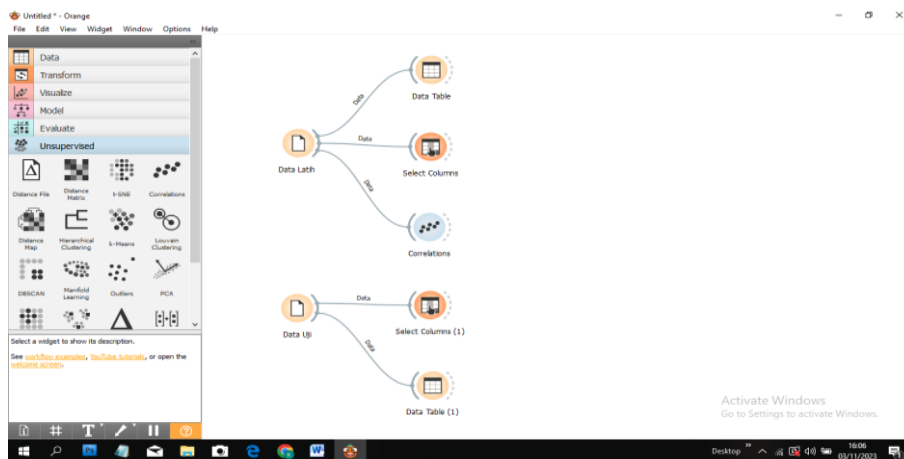
Digunakan untuk menyusun domain data secara manual. Pengguna dapat memutuskan atribut mana yang akan digunakan dan bagaimana caranya *orange* membedakan atribut biasa, atribut kelas (opsional), dan atribut meta. Misalnya untuk membangun model klasifikasi, domain akan terdiri dari sekumpulan atribut dan atribut kelas diskrit.



Gambar 4.4 Tampilan widget *select columns*

4.6 Tampilan Widget *Correlations*

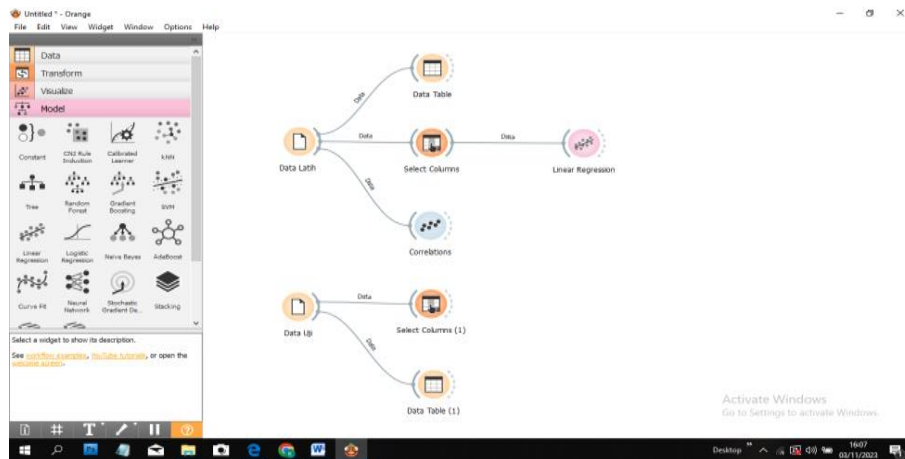
Menghitung skor korelasi berpasangan (*pearson*) atau berpasangan (*spearman*) untuk semua pasangan fitur dalam kumpulan data. Metode ini hanya dapat mendeteksi hubungan yang monoton. Daftar pasangan atribut dengan koefisien korelasi.



Gambar 4.5 Tampilan widget *correlations*

4.7 Tampilan *Widget Linear Regression*

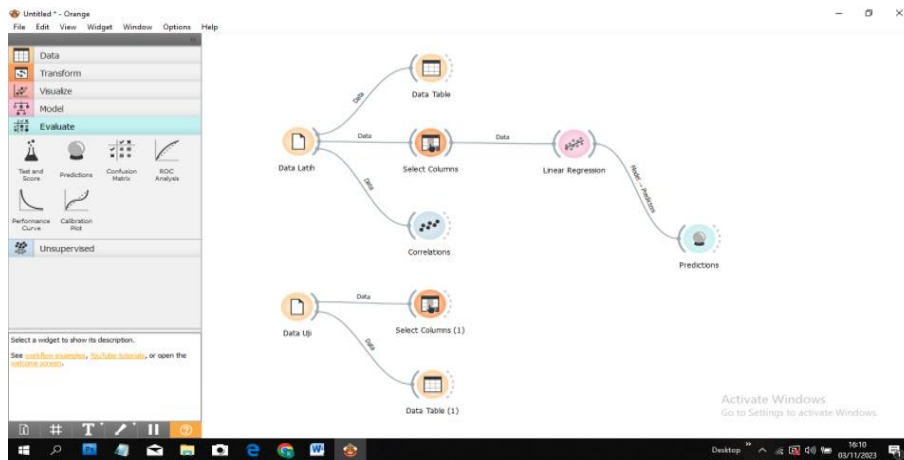
Membuat pelajar/prediktor yang mempelajari fungsi linear dari data masukannya. Model dapat mengidentifikasi hubungan antara prediktor x_i dan variabel respon y . *Linear regression* menggunakan *processing default* ketika tidak ada *preprocessor* lain yang diberikan. Untuk mengamati koefisien linear regresi pertama bangun model dan kemudian kirimkan model tersebut ke data tabel akan secara otomatis menghubungkan keluaran koefisien ke data tabel dan mengamati variabel mana yang berkorelasi positif dan negatif dengan prediksi.



Gambar 4.6 Tampilan *widget linear regression*

4.8 Tampilan *Widget Predictions*

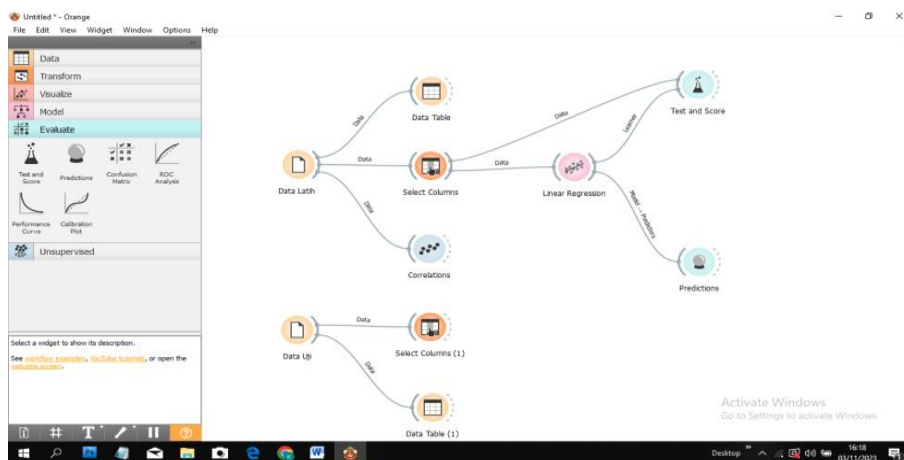
Menampilkan model prediksi pada data dan menerima kumpulan data dan satu atau lebih prediktor. *Widget* menunjukkan probabilitas dan keputusan akhir dari model prediksi. Keluaran *widget* adalah kumpulan data lain di mana prediksi di tambah sebagai atribut meta baru.



Gambar 4.7 Tampilan *widget predictions*

4.9 Tampilan *Widget Test and Score*

Pada Gambar 4.8 menguji *algoritma* pembelajaran. Skema pengambilan sampel yang berbeda tersedia. Termasuk menggunakan data uji terpisah. Pengklasifikasi yang berbeda seperti akurasi klasifikasi dan area di bawah kurva, untuk menampilkan hasil evaluasi yang dapat digunakan oleh *widget* lain untuk menganalisis kinerja pengklasifikasi, seperti ROC (*Receiver Operating Characteristic*) Analisis atau *Confusion Matrix*.



Gambar 4.8 Tampilan *widget test and score*