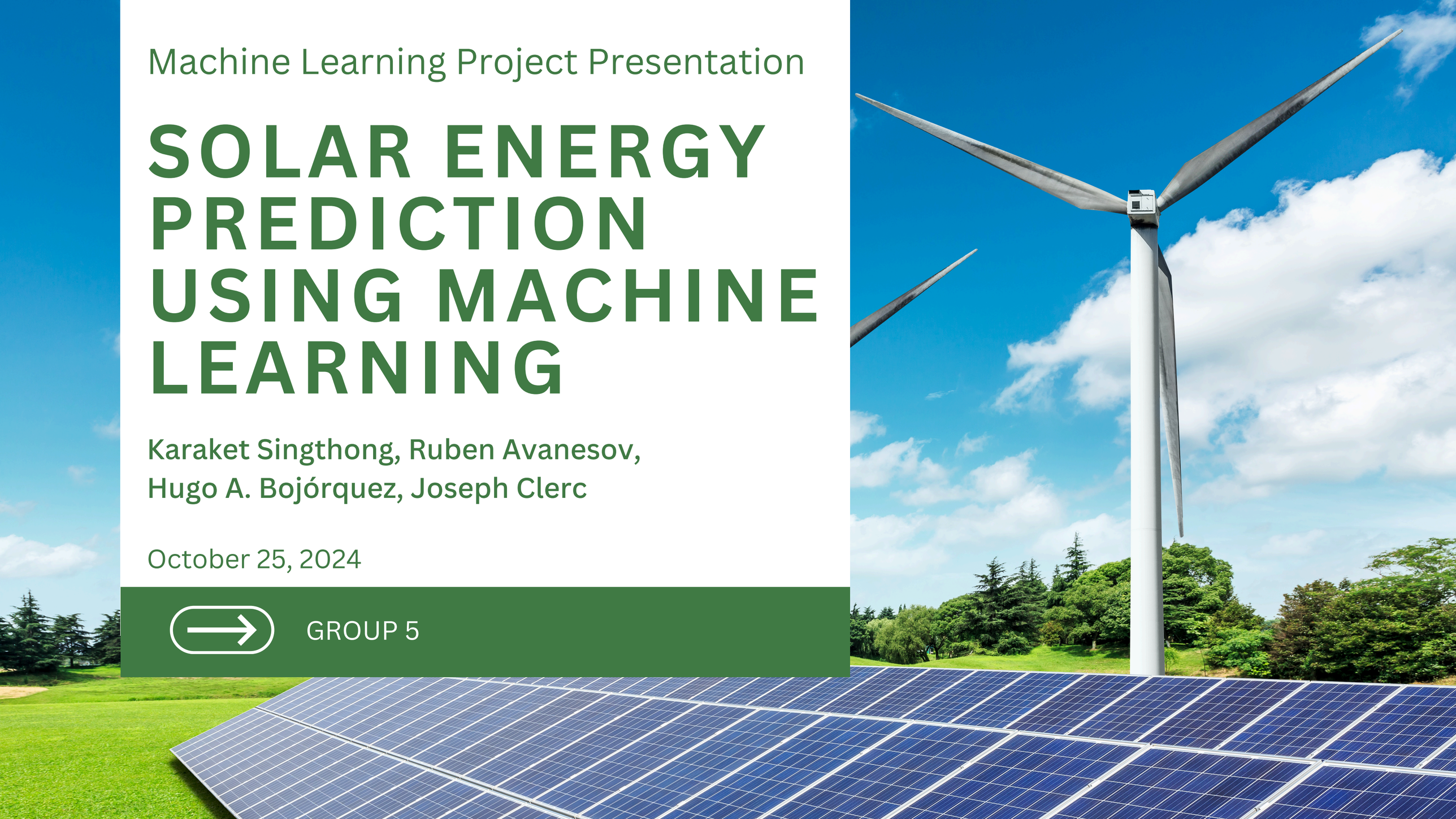
Machine Learning Project Presentation

SOLAR ENERGY PREDICTION USING MACHINE LEARNING

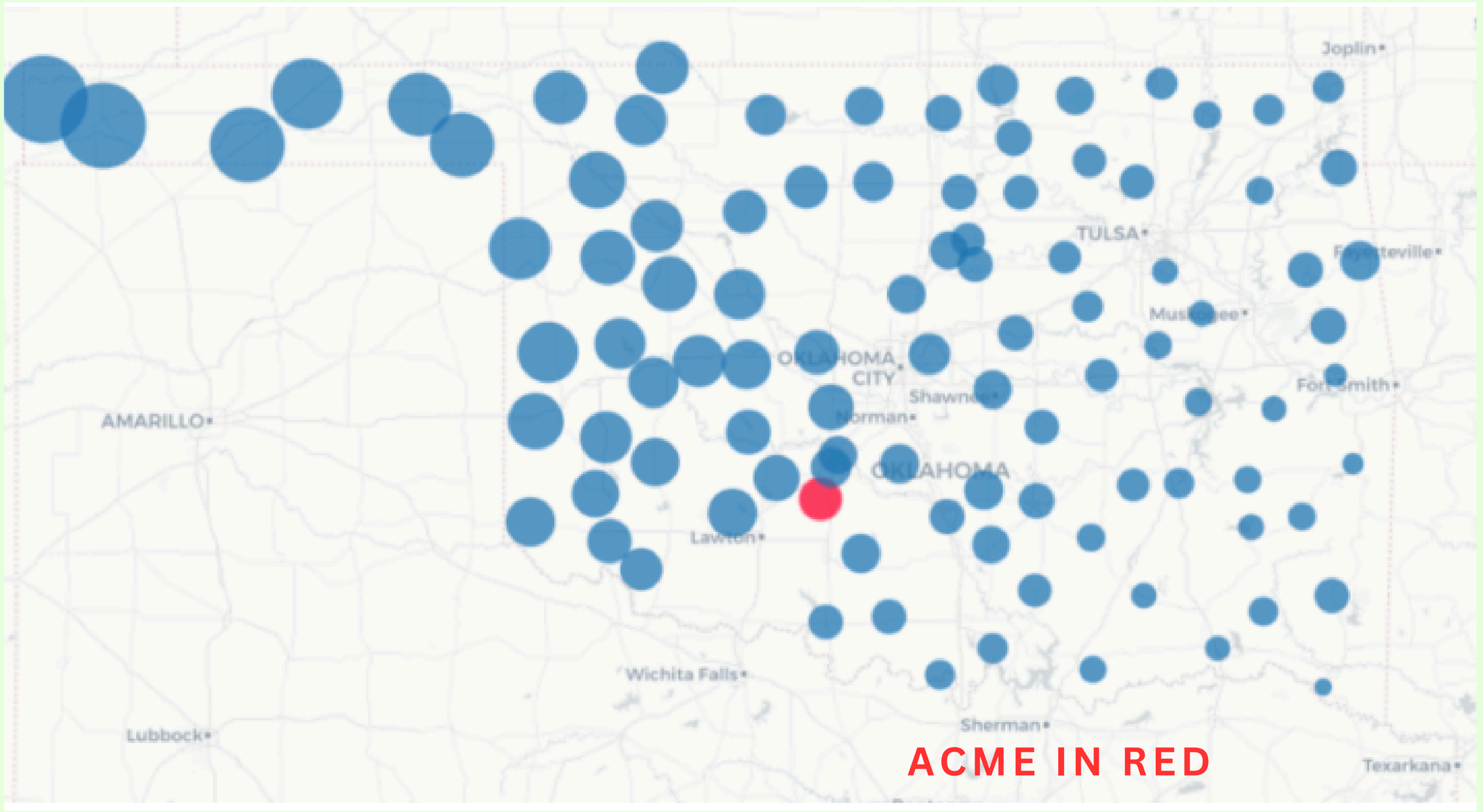
Karaket Singthong, Ruben Avanesov,
Hugo A. Bojórquez, Joseph Clerc

October 25, 2024

GROUP 5



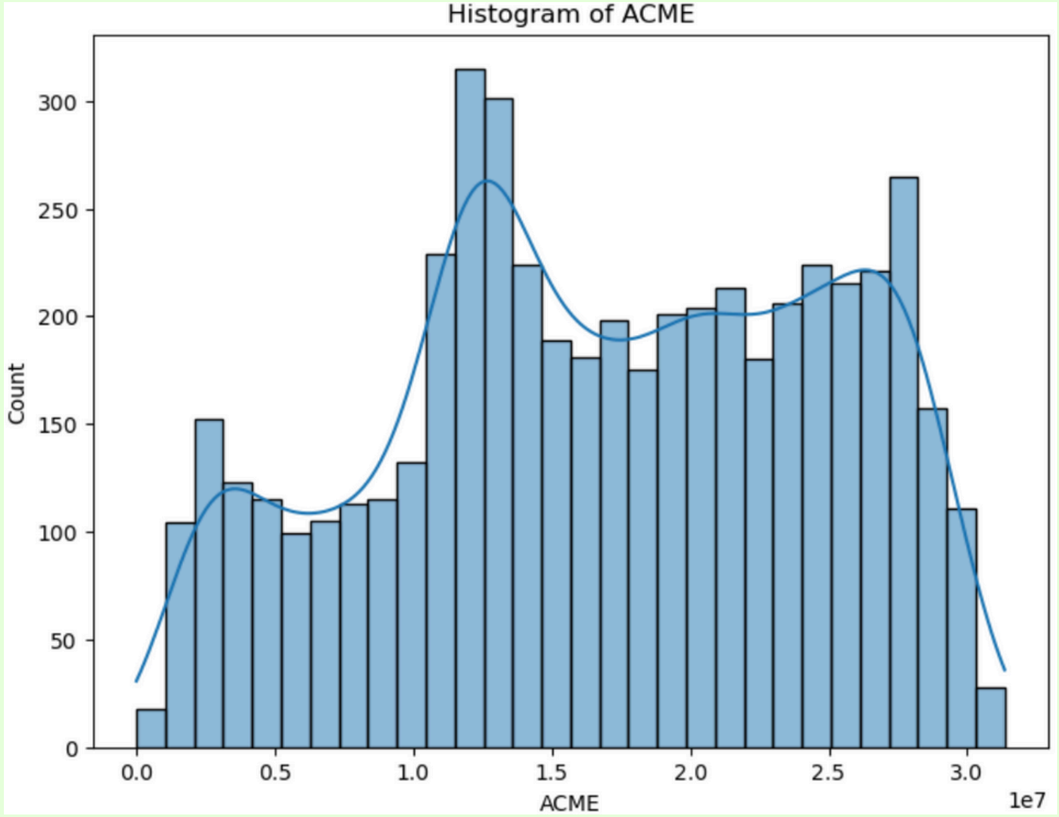
LOCATION OF ACME

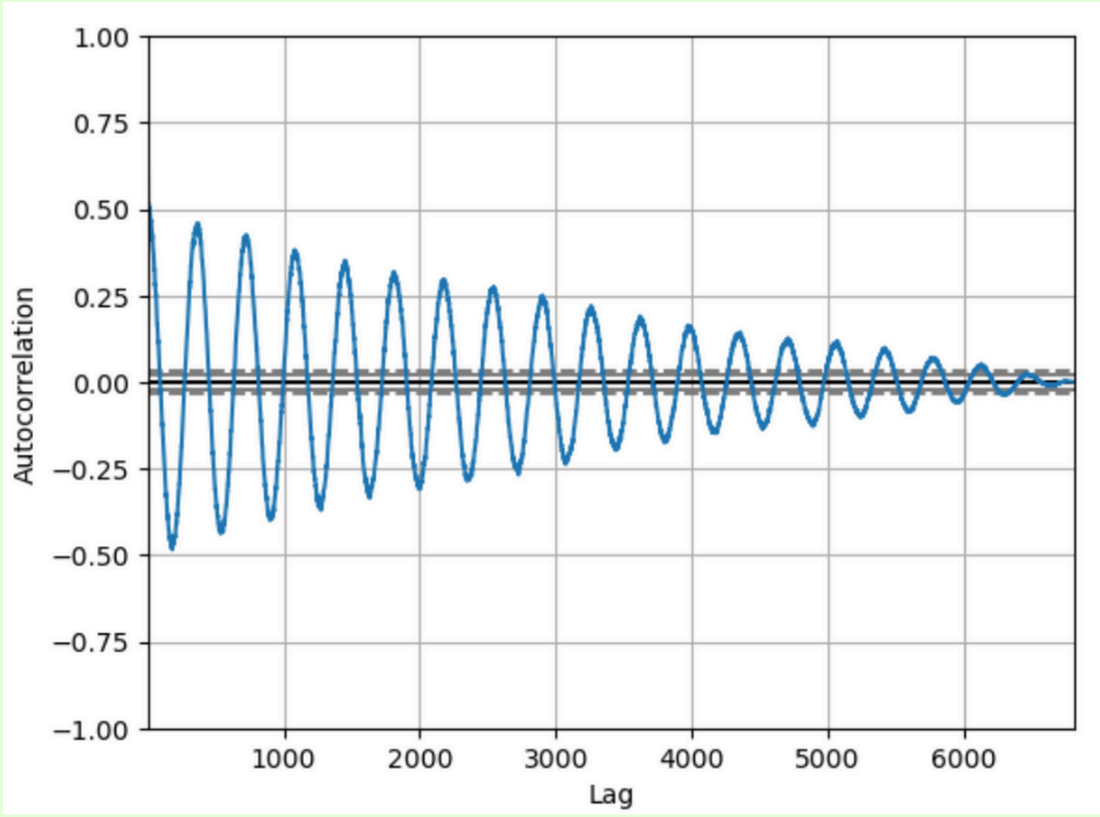


(Built using station_info.csv)

SOME STATISTICS CONCERNING ACME

There is potential for autoregression. Also, the model is not normally distributed





FEATURE ENGINEERING

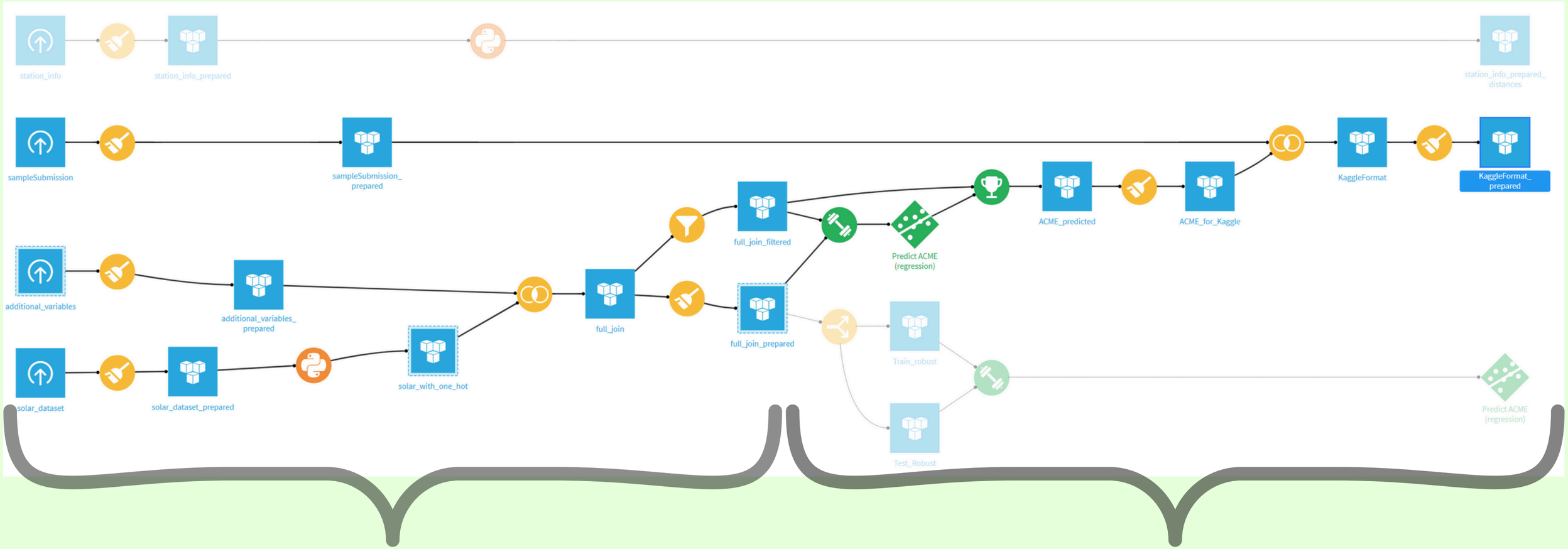
Correlations to ACME of the new variables

	ACME	month_12	ACME_rolli...	ACME_rolli...	ACME_med...	ACME_med...	month_2	month_3	month_4	month_5	month_6	month_7	month_8	month_9	month_10	month_11
ACME	1.000	-0.323	0.772	0.774	0.733	0.717	-0.161	-0.024	0.125	0.190	0.260	0.328	0.195	0.053	-0.118	-0.253
month_12	-0.323	1.000	-0.404	-0.380	-0.439	-0.457	-0.086	-0.091	-0.089	-0.091	-0.089	-0.091	-0.091	-0.089	-0.091	-0.089
ACME_rollin...	0.772	-0.404	1.000	0.951	0.931	0.901	-0.194	-0.021	0.172	0.236	0.359	0.408	0.222	0.024	-0.147	-0.326
ACME_rollin...	0.774	-0.380	0.951	1.000	0.894	0.867	-0.198	-0.026	0.165	0.214	0.337	0.390	0.231	0.047	-0.145	-0.305
ACME_med...	0.733	-0.439	0.931	0.894	1.000	0.958	-0.203	-0.041	0.143	0.249	0.365	0.457	0.260	0.053	-0.152	-0.336
ACME_med...	0.717	-0.457	0.901	0.867	0.958	1.000	-0.199	-0.041	0.173	0.250	0.363	0.467	0.252	0.042	-0.177	-0.303
month_2	-0.161	-0.086	-0.194	-0.198	-0.203	-0.199	1.000	-0.089	-0.087	-0.089	-0.087	-0.089	-0.089	-0.087	-0.089	-0.087
month_3	-0.024	-0.091	-0.021	-0.026	-0.041	-0.041	-0.089	1.000	-0.092	-0.093	-0.092	-0.093	-0.093	-0.092	-0.093	-0.091
month_4	0.125	-0.089	0.172	0.165	0.143	0.173	-0.087	-0.092	1.000	-0.092	-0.090	-0.092	-0.092	-0.090	-0.092	-0.090
month_5	0.190	-0.091	0.236	0.214	0.249	0.250	-0.089	-0.093	-0.092	1.000	-0.092	-0.093	-0.093	-0.092	-0.093	-0.091
month_6	0.260	-0.089	0.359	0.337	0.365	0.363	-0.087	-0.092	-0.090	-0.092	1.000	-0.092	-0.092	-0.090	-0.092	-0.090
month_7	0.328	-0.091	0.408	0.390	0.457	0.467	-0.089	-0.093	-0.092	-0.093	-0.092	1.000	-0.093	-0.092	-0.093	-0.091
month_8	0.195	-0.091	0.222	0.231	0.260	0.252	-0.089	-0.093	-0.092	-0.093	-0.092	-0.093	1.000	-0.092	-0.093	-0.091
month_9	0.053	-0.089	0.024	0.047	0.053	0.042	-0.087	-0.092	-0.090	-0.092	-0.090	-0.092	-0.092	1.000	-0.092	-0.090
month_10	-0.118	-0.091	-0.147	-0.145	-0.152	-0.177	-0.089	-0.093	-0.092	-0.093	-0.092	-0.093	-0.093	-0.092	1.000	-0.091
month_11	-0.253	-0.089	-0.326	-0.305	-0.336	-0.303	-0.087	-0.091	-0.090	-0.091	-0.090	-0.091	-0.091	-0.090	-0.091	1.000

+ 0 -

In order to be able to use the rolling values for the predictions, we calculate their averages over the available years to extrapolate over the unseen data

DATAIKU FLOW



Data Preparation

Prediction Generation

DATA PREPROCESSING

1 .Data Upload

Uploaded all datasets and sample submission files to Dataiku for preprocessing.

2 . Station_info.csv

- Added a GeoPoint column for station visualization on a map.
- Calculated distances between stations and ACME for potential proximity-based predictions.

3 . Solar_dataset.csv

- Removed data for irrelevant solar stations.
- Parsed the date column and added categorical month values using one-hot encoding.
- Calculated 30-day and 7-day rolling medians for the 'ACME' column, filling missing values with group means.
- Aggregated median 'ACME' values for each MonthDay and week-month combination across all years.

4. Additional_variables.csv

- Parsed the date column and filled empty cells with zeros.
- Lagged all columns by one day to account for temporal dependencies (gave us better correlations).

5 .Dataset Joining

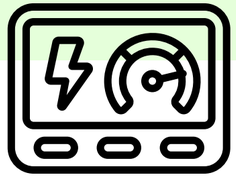
Joined Solar and additional variables datasets on the date column.

6 .Data Splitting

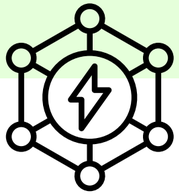
- Separated data where ACME values were non-empty for model training.
- Split the data 70/30 into training and testing sets.

OTHER PREPROCESSING TRANSFORMATIONS THAT WEREN'T IMPLEMENTED

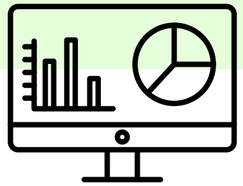
The previous slide is what we ended up doing, however in the meantime we considered a lot of different options to preprocess the data, none of which are implemented



Dimensionality reduction on the "additional variables" (didn't help the model)



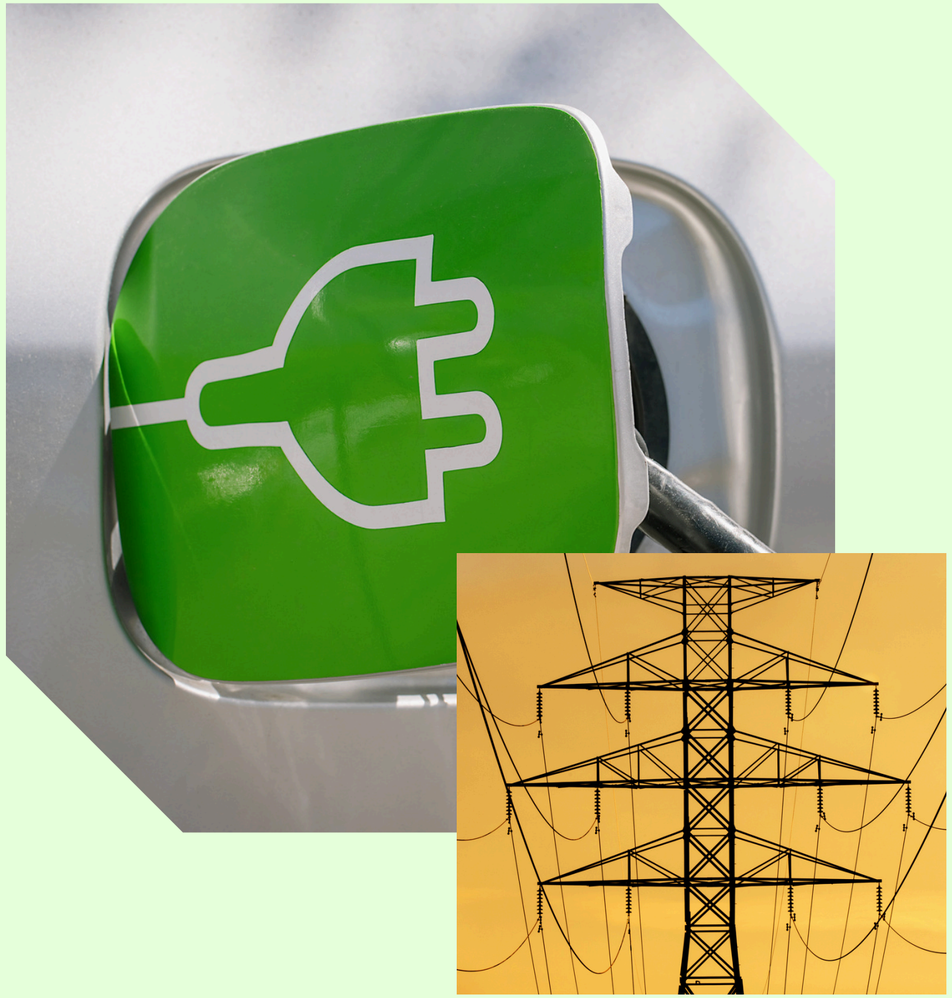
Filling in the missing the missing variables in the additional variables with predictions rather than zeros (also didn't help predictions)



Dimensionality reduction on the "PC…" variables. (doesn't work, it is simply better to just delete some of them)

Time series decomposition (we couldn't figure it out so we just used medians of same dates, and projected aggregated medians, and one hot encoded months)

MODELING APPROACH



LIST THE MACHINE LEARNING MODELS WE TESTED!!

- Linear Regression
- Gradient boosted trees
- XGBoost

FINALLY WE ENDED UP WITH :)

XGBoost : Provided the best results and the most convenient training times

KEY HYPERPARAMETERS AND TUNING STRATEGIES

max n trees: 1000
max depth: 10
learning rate: 0.01
Min sum of instance weight in a child: 2.75
Subsample ratio of the training instance: 0.5
Early stopping turned on, and stopping rounds: 5

We trained a lot of models and reused what worked with each implementation

RESULTS

BENCHMARK MODEL

MAE: 3'060'000

BENCHMARK *Started Today at 17:49 , ended Today at 17:49* *1 model* *473 / 474 features*

Ordinary Least Squares (benchmark) 3.06e+6 Done 1 minute ago (2024-10-25 17:49:46)

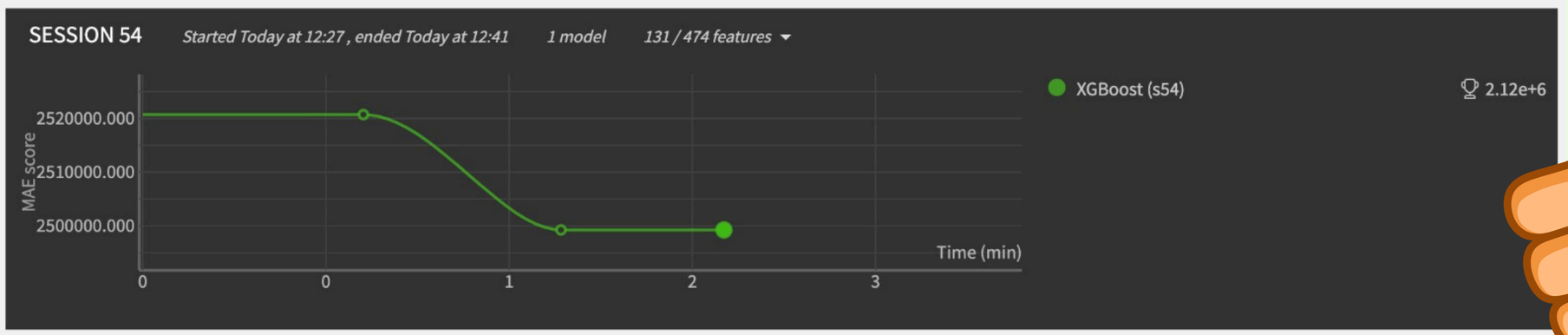
Time variable **Date**

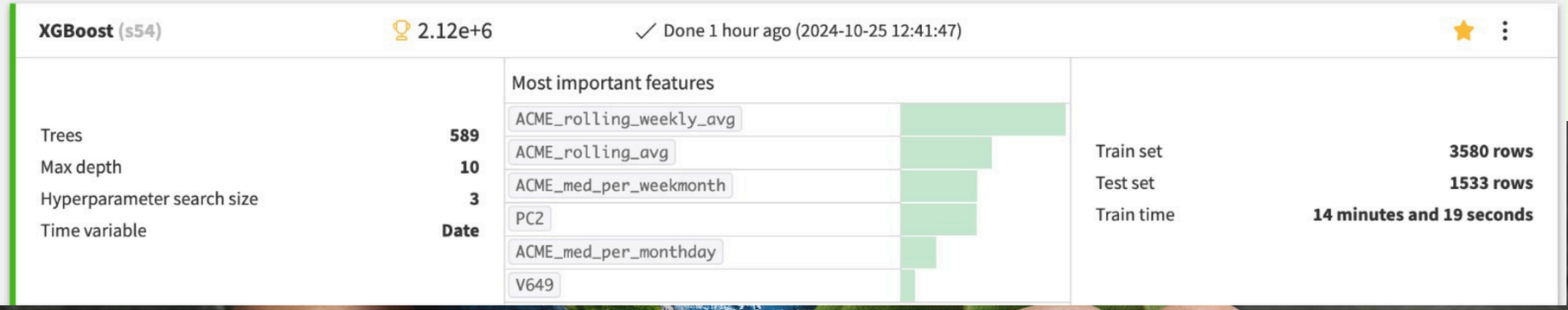
Top coefficients

NUM_DERIVATIVE:V1800^2	↘	★★★
NUM_DERIVATIVE:V361^2	↗	★★☆
NUM_DERIVATIVE:sqrt(V1081)	↘	★★★
NUM_DERIVATIVE:sqrt(V1369)	↗	★★☆
NUM_DERIVATIVE:V1080^2	↗	★★☆
NUM_DERIVATIVE:V1081^2	↘	★★☆

Train set	**3580 rows**
Test set	**1533 rows**
Train time	**about 7 seconds**

STATISTICS OF OUR BEST MODEL





MAE: 2'120'000

KEY POINTS

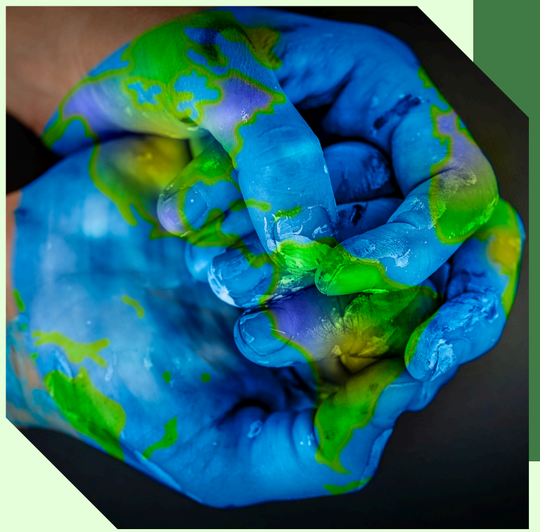
- We trained the model on the train/test split initially, in order to tune the hyperparameters and find the best model.
- In the end however, we retrain the model on the entire train + test data in order to make the final predictions.

- We do not use Feature reduction as it does not better our results
- We then output the results of the prediction of our best model. We found that the predictions suffer from underprediction and as such we multiply them by 1.07 in order to optimize our Kaggle result.
- We join the predictions to the Kaggle sample submission format in order to submit to Kaggle

- In the feature handling, we remove all the “PC…” variables after PC15 because they have low correlation to ACME. The rest is preserved
- We create some interaction terms that proved to have high correlation in our preliminary python research:

Interaction between	PC6	and	PC9
Interaction between	V1800	and	V6265
Interaction between	V5401	and	V7834
Interaction between	PC12	and	PC13
Interaction between	PC14	and	PC15
Interaction between	V649	and	V2520
Interaction between	V2074	and	V2520
Interaction between	V1800	and	V649
Interaction between	V649	and	V7145

THOUGHTS FOR FUTURE PROGRESSION



Combining Model Predictions using a third model

- We thought it could have been interesting to make predictions using a time series model separately in order to then use another model to combine it with the predictions of our gradient boosting model. However, we could not figure out how to do it in Dataiku.
- We also tried creating a model that would predict when outliers would happen, but couldn't get robust results

Transform the Gradient Boosting predictions

- We could have also tried to transform the predictions of our model in order to match the frequencies of the quantiles across ACME, but here again, it is difficult to figure out how to do this within the built-in dataiku functions

THANK YOU