WIND SHEAR CASE STUDY

Analysis & Results

Abstract

As part of the GE Analytics Engineer Certification, the Case Study: Virtual Wind Shear Sensor was analyzed to answer the feasibility of using load sensor data to estimate wind shear, and make recommendations if GE Wind could launch an NPI program to offer a "Virtual Wind Shear Sensor" using the data collected from the load sensors on its next generation of Wind Turbines.

Table of Contents

List of Figures

List of Tables

Introduction

As part of the GE Analytics Engineer Certification, the ***Case Study: Virtual Wind Shear Sensor*** was analyzed to answer the feasibility of using load sensor data to estimate wind shear, and make recommendations if GE Wind could launch an NPI program to offer a "Virtual Wind Shear Sensor" using the data collected from the load sensors on its next generation of Wind Turbines.

Load sensors are installed on every wind turbine at the hub to sense the nodding, pitching, yawing, and other loading parameters. Some of them are also influenced by wind shear. For further details, read ***"Wind Shear Case Study.docx"*** document.

This case study includes a data set collected from one wind turbine that gives both wind speed from a 5 sensor met mast and load parameters from the load sensors.

Basically, this analysis follows the CRISP-DM[1] phases, as it is illustrated in the following picture

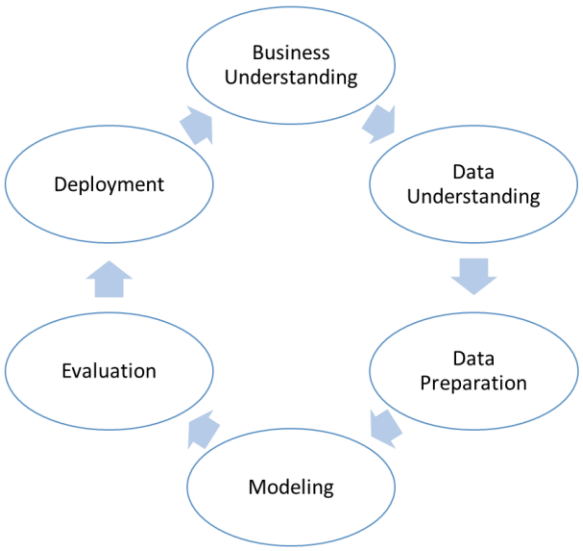


FIGURE 1 CRISP-DM MAJOR PHASES

Business Understanding.

There are three goals to accomplish with this analysis.

1. Develop and evaluate a ***classifier*** to see how well a ***speed profile*** can be determined from the load sensor data. Use load sensors data (X – 14 variables) and classify the ***wind speed profiles*** (Y – ShearTypeClass) as follows:

[1] Cross Industry Standard Process for Data Mining (https://en.wikipedia.org/wiki/Cross_Industry_Standard_Process_for_Data_Mining)

0: wind speed profile follows a power law;
1: LLJ
2: Flat
3: Others

2. Create a predictive model for $\boldsymbol{\alpha}$ from the load sensor data and determine how well it estimates the actual $\boldsymbol{\alpha}$**.** If the speed profile follows a power law (ShearTypeClass = 0), the ***shear* ($\boldsymbol{\alpha}$)** can be calculated by using wind speeds at altitudes of **38m** and **78.7m**. See following equation:

$$V_{38m} = V_{78.7m}\left(\frac{38}{78.7}\right)^{\alpha}$$

$$\alpha = \frac{ln\left(\frac{V_{38m}}{V_{78.7m}}\right)}{\ln\left(\frac{38}{78.7}\right)}$$

3. For speed profiles labeled as 1, 2, and 3, how might you create a model to estimate these profiles?

Good to know

The **wind profile power law** is a relationship between the wind speeds at one height, and those at another. The **power law** is often used in wind power assessments where wind speeds at the height of a turbine (>~ 50 metres) must be estimated from near surface wind observations (~10 metres), or where wind speed data at various heights must be adjusted to a standard height prior to use.

The wind profile power law relationship is:

$$V(h) = V(HH) * \left(\frac{h}{HH}\right)^{\alpha}$$

where $V(h)$is the wind speed (in metres per second) at altitude (height) h (in metres), $V(HH)$ is the wind speed at altitude (reference height) HH, HH is hub height (it is also called the *roughness coefficient* length and is expressed in metres, and which depends basically on the land type, spacing and height of the roughness factor (water, grass, etc.)). α is the power (i.e. wind shear). The exponent α is an empirically derived coefficient that varies dependent upon the stability of the atmosphere. For neutral stability, α is approximately 1/7 [2](0.143).

Factors influencing wind speeds[3]

Empirical evidence has shown that at a great height over the ground surface (in the region of one kilometre) the land surface influence on the wind is negligible. However, in the lowest atmospheric layers the wind speed is affected by ground surface friction factors.

[2] Wind profile power law: (https://en.wikipedia.org/wiki/Wind_profile_power_law)
[3] Methodologies Used in the Extrapolation of Wind Speed Data (http://www.intechopen.com/books/wind-farm-technical-regulations-potential-estimation-and-siting-%20assessment/methodologies-used-in-the-extrapolation-of-wind-speed-data-at-different-heights-and-its-impact-in-th)

There are two well-defined factors affecting wind speed: *environmental factors*, ranging from local topography, weather to farming crops, etc. and *artificial factors* ranging from man-made structures to permanent and temporary hindrances such as buildings, houses, fences and chimneys.

The *friction coefficient* α is set empirically and the equation can be used to adjust the data reasonably well in the range of 10 up to 100-150 metres.

Landscape type	Friction coefficient α
Lakes, ocean and smooth hard ground	0.10
Grasslands (ground level)	0.15
Tall crops, hedges and shrubs	0.20
Heavily forested land	0.25
Small town with some trees and shrubs	0.30
City areas with high rise buildings	0.40

TABLE 1: FRICTION COEFFICIENT A FOR A VARIETY OF LANDSCAPES.

Wind Shear[4]

The wind speed profile trends to a lower speed as we move closer to the ground level. This is designed as wind shear.
The wind speed at a certain height above ground can be estimated as a function of height above ground z and the roughness length Z0 from table 2 in the current wind direction from the formula:

$$V(z) = V_{ref} \frac{ln\left(\frac{z}{Z_0}\right)}{\ln\left(\frac{z_{ref}}{Z_0}\right)}$$

The reference V_{ref} is a known wind speed at a reference height z_{ref}.
The formula assumes neutral atmospheric stability conditions under which the ground surface is neither heated nor cooled compared with the air temperature.

Inside of a Wind Turbine[5]

This web site gives an overall about the wind turbine components, and how they work.

[4] "Wind Shear, Roughness Classes and Turbine Energy Production". M. Ragheb. 4/2/2015.
[5] THE INSIDE OF A WIND TURBINE (http://www.energy.gov/eere/wind/inside-wind-turbine-0)

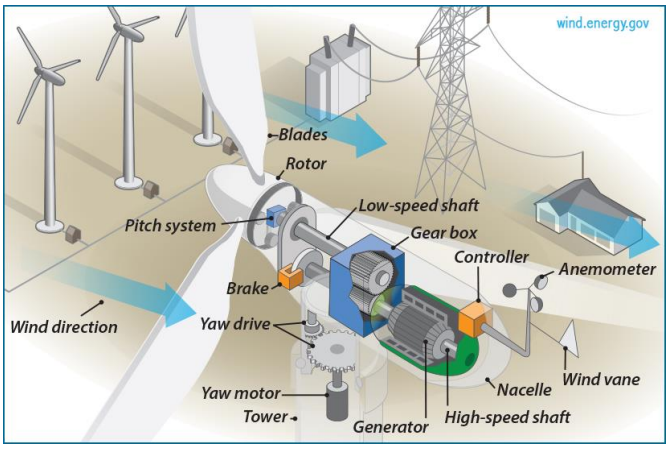


FIGURE 2 WIND TURBINE COMPONENTS

Applications

In order to analyze the data and program statistics models, the following applications were used:

Application Name	Brief Description and Usage
Tableau	It is a data visualization software. It connects easily to nearly any data source. It is used to do histograms, box-plots and scatter charts. Download: http://www.tableau.com/products/trial
Excel	It is a spreadsheet software. It is used to delve into data details, and see if some effort is requested to fix data quality.
R	It is a programming language and software environment for statistical computing and graphics. It is used to some statistics models and correlation analysis. Download R: https://cran.rstudio.com/ Download RStudio: https://www.rstudio.com/products/rstudio/download3/
KNIME	It is an open source data analytics, reporting and integration platform. It integrates various components for machine learning and data mining. It is used to obtain stats, apply statistics models, data cleaning/manipulation, select best model, etc. Download: https://www.knime.org/downloads/overview

Data Understanding

The data set provided has the following columns (it is important to notice the data type for each column):

Column Name	Description	Data Type
datetime	Date when data was taken	Date / Time
m38	wind speed at height of 38 m, m/sec	Numeric
m58	wind speed at height of 58 m, m/sec	Numeric

m78	wind speed at height of 78 m, m/sec	Numeric
m103	wind speed at height of 103 m, m/sec	Numeric
m122	wind speed at height of 122 m, m/sec	Numeric
RPM_0P	0P component of RPM in 1/min	Numeric
nodd_0P	0P component of nodding moment in Nm	Numeric
nodd_3C	3P (cosine) component of nodding moment in Nm	Numeric
nodd_3S	3P (sine) component of nodding moment in Nm	Numeric
pitch_d_0P	0P component of d-component of pitch angle	Numeric
pitch_q_0P	0P component of q-component of pitch angle	Numeric
pitch_d_3C	3P (cosine) component of d-component of pitch angle	Numeric
pitch_d_3S	3P (sine) component of d-component of pitch angle	Numeric
yaw_0P	0P component of yawing moment in Nm	Numeric
yaw_3C	3P (cosine) component of yawing moment in Nm	Numeric
yaw_3S	3P (sine) component of yawing moment in Nm	Numeric
P_el	0P component of electrical power in kW	Numeric
V_estim	0P component of MBC estimated wind speed in m/s	Numeric
pitch_col_0P	0P component of collective pitch angle	Numeric
ShearTypeClass	0: wind speed profile follows a power law 1: LLJ (Low-level Jet) 2: Flat 3: Others	Nominal

TABLE 2 DATA SET PROVIDED

Exploration of data

Here are important statistics about the data set.

ShearTypeClass	Number of Records	% of Total Number of Records
Power Law	6,346	78.87%
LLJ	1,177	14.63%
Flat	394	4.90%
Others	129	1.60%
# Total of Records	**8,046**	**100.00%**

TABLE 3 NUMBER OF RECORDS BY SHEARTYPECLASS

We can see that **78.87%** (around 80%) of records were classified as “Power Law”. About 80% of the wind speed profile follows a power law”, followed by LLJ (Low-level Jet) with **14.63%**.

Also, data set contains data collected during **12 days** (from 6/17/2015 to 6/28/2015). Next chart illustrates how the number of records by ShearTypeClass is split by day.

Shear Type ..	17	18	19	20	21	22	23	24	25	26	27	28
Power Law	398	697	696	696	681	676	691	685	682	222	136	86
LLJ	1	8	14	4	14	3	19	17	25	376	424	272
Flat	7	5	9	18	21	41	10	17	8	82	140	36
Others		10	1	2	4			1	5	40	20	46
Grand Total	**406**	**720**	**720**	**720**	**720**	**720**	**720**	**720**	**720**	**720**	**720**	**440**

FIGURE 3 NUMBER OF RECORDS PER DAY BY SHEARTYPECLASS

Notice that most of "Power Law" records were collected from day **18th to 25th**. However, analyzing the total number of records per day, only the days 17th and 28th have around 400 of records (406 and 440 respectively), and the rest of the days the number total of records per day look equal (720 records). Even though days 26th and 27th have same number of records versus other days, these two days have low number of records for Power Law and have a more records for LLJ, Flat and Others. See table below.

Shear Type Class	17	18	19	20	21	22	23	24	25	26	27	28
Power Law	398	697	696	696	681	676	691	685	682	222	136	86
LLJ	1	8	14	4	14	3	19	17	25	376	424	272
Flat	7	5	9	18	21	41	10	17	8	82	140	36
Others		10	1	2	4			1	5	40	20	46
Grand Total	406	720	720	720	720	720	720	720	720	720	720	440

TABLE 4 NUMBER OF RECORDS PER DAY BY SHEARTYPECLASS.

Days 17th and 28th are not complete, it means only data were collected during few hours on both days. In fact, 17th only includes 14 hours (from 10 to 23 hours) and 28th includes data for 15 hours (from 0 to 14 hours). See below table.

			Shear Type Class		
Day	**Flat**	**LLJ**	**Others**	**Power Law**	**# Hours / Day**
17	2	1		14	**14**
18	3	7	7	24	**24**
19	5	6	1	24	**24**
20	8	4	2	24	**24**
21	5	7	3	24	**24**
22	11	3		24	**24**
23	6	6		24	**24**
24	8	8	1	24	**24**
25	5	11	4	24	**24**
26	11	21	12	22	**24**
27	14	23	7	19	**24**
28	7	15	8	12	**15**

TABLE 5 NUMBER OF HOURS PER DAY HAVING DATA BY SHEARTYPECLASS

Data Preparation

In addition to data set columns provided, the following columns were added during the analysis in order to simply the data analysis.

Column Name	Description
ShearTypeClassROCDesc	Power Law (for those records having ShearTypeClass = 0) No Power Law (ShearTypeClass <> 0)
ShearTypeClassROCInt	1: Power Law 0: No Power Law
ShearTypeClassDesc	ShearTypeClass field is converted to ordinal as follows: Power Law (ShearTypeClass = 0) LLJ (ShearTypeClass = 1) Flat (ShearTypeClass = 2) Others (ShearTypeClass = 3)
datetime_time	datetime converted to date/time field
MyYearData	Year of datetime
MyMonthData	Month of datetime
MyDayData	Day of datetime
MyHourData	Hour of datetime
data_date_int	datetime in format YYYYMMDD
data_date_hr_int	datetime in format YYYYMMDDHH
data_date_int_str	data_date_int in string format
shear_a	wind shear calculation using wind speeds at altitudes of 38m and 78.7m. **Note: it is a continuous to be predict.**

TABLE 6 ADDITIONAL COLUMNS TO ORIGINAL DATA SET

Column	Min	Max	Mean	Std. Dev.	Variance	Skewness	Kurtosis	Overall Sum	No. Missing	No. NaNs	No. +∞	No. -∞	Median	Row Count
RPM_0P	7.961	14.618	13.447	1.548	2.396	-1.886	2.706	108,192.888	0	0	0	0	14.312	8,046
nodd_0P	-1,593.100	205.970	-612.926	296.029	87,633.172	0.102	0.720	-4,931,605.498	0	0	0	0	-691.715	8,046
nodd_3C	-267.870	652.530	48.305	94.718	8,971.441	1.819	5.740	388,665.754	0	0	0	0	27.957	8,046
nodd_3S	-183.160	561.630	93.961	93.754	8,789.734	1.151	2.716	756,010.338	0	0	0	0	78.897	8,046
pitch_d_0P	-1.586	2.433	0.414	0.767	0.588	0.862	0.030	3,333.607	0	0	0	0	0.000	8,046
pitch_q_0P	-1.440	1.493	0.112	0.244	0.059	0.574	3.452	902.926	0	0	0	0	0.003	8,046
pitch_d_3C	-0.597	0.188	-0.076	0.103	0.011	-1.353	1.645	-615.329	0	0	0	0	-0.030	8,046
pitch_d_3S	-0.547	0.354	-0.001	0.050	0.002	-0.466	9.966	-5.073	0	0	0	0	0.000	8,046
yaw_0P	-387.520	489.130	58.314	84.087	7,070.544	0.511	1.355	469,192.720	0	0	0	0	27.151	8,046
yaw_3C	-201.440	487.160	57.116	80.856	6,537.654	1.136	2.666	459,558.117	0	0	0	0	42.674	8,046
yaw_3S	-274.140	208.730	-43.513	52.226	2,727.559	-0.512	2.375	-350,102.409	0	0	0	0	-38.360	8,046
P_el	3.384	2,359.400	1,592.617	676.685	457,902.860	-0.468	-1.129	12,814,197.957	0	0	0	0	1,770.100	8,046
V_estim	2.756	16.457	9.584	2.236	5.000	-0.064	-0.392	77,113.962	0	0	0	0	9.793	8,046
pitch_col_0P	0.040	15.102	2.592	2.885	8.322	1.420	1.643	20,851.847	0	0	0	0	1.740	8,046

TABLE 7 STATISTICS OF LOAD SENSORS DATA VARIABLES (ALL DAYS)

From the previous table, we can notice some important aspects for each load sensor variables:

- No NULL data in all sensor data variables.
- The skewness (symmetry in a distribution) is not zero, which means they are not having normal distribution.
- The kurtosis for some variables are bigger than 3, then the dataset for that variable has heavier tails than normal distribution.
- The variances are big for some variables. ***The variance could be affected by the mix of shear types***.
- Notice in most of the cases, MEAN and MEDIAN are distant, which reflects distributions have skewness.

What will be the behavior of these statistics if we ***remove the 4 days*** (days 17, 26, 27, 28) where data seems to be not completed for Power Law shear type? In general, it is the same behavior, some stats become better and some others become worst. See below table.

Column	Min	Max	Mean	Std. Dev.	Variance	Skewness	Kurtosis	Overall Sum	No. Missing	No. NaNs	No. +∞	No. -∞	Median	Row Count
RPM_0P	7.999	14.567	13.515	1.489	2.218	-1.972	3.017	77,845.272	0	0	0	0	14.313	5,760
nodd_0P	-1,593.100	205.970	-603.522	305.226	93,162.995	0.019	0.626	-3,476,288.623	0	0	0	0	-690.835	5,760
nodd_3C	-267.870	435.110	23.020	58.154	3,381.871	0.464	2.675	132,594.545	0	0	0	0	18.908	5,760
nodd_3S	-183.160	413.400	75.586	73.369	5,382.954	0.541	1.806	435,375.232	0	0	0	0	70.491	5,760
pitch_d_0P	-1.586	2.433	0.361	0.757	0.574	0.861	0.098	2,077.241	0	0	0	0	0.000	5,760
pitch_q_0P	-1.440	1.493	0.095	0.251	0.063	0.482	3.701	544.642	0	0	0	0	0.002	5,760
pitch_d_3C	-0.597	0.188	-0.067	0.084	0.007	-1.039	1.113	-387.062	0	0	0	0	-0.034	5,760
pitch_d_3S	-0.420	0.354	-0.001	0.051	0.003	-0.472	7.703	-5.964	0	0	0	0	0.000	5,760
yaw_0P	-387.520	489.130	54.879	83.816	7,025.090	0.513	1.851	316,105.420	0	0	0	0	27.526	5,760
yaw_3C	-201.440	308.760	37.285	58.716	3,447.577	0.439	1.470	214,759.208	0	0	0	0	32.852	5,760

yaw_3S	-274.140	208.730	-33.299	43.392	1,882.892	0.079	3.419	-191,803.591	0	0	0	0	-32.996	5,760
P_el	19.008	2,359.400	1,617.609	659.190	434,531.310	-0.534	-1.024	9,317,426.213	0	0	0	0	1,794.550	5,760
V_estim	2.826	16.170	9.674	2.198	4.830	-0.050	-0.381	55,721.755	0	0	0	0	9.845	5,760
pitch_col_0P	0.040	14.541	2.675	2.917	8.512	1.401	1.485	15,406.625	0	0	0	0	1.791	5,760

TABLE 8 STATISTICS OF LOAD SENSORS DATA VARIABLES (WITHOUT DAYS 17, 26, 27, 28)

Following 4 charts show histograms for each variable. Notice in red, how many values are ZERO or too close to ZERO. They include all data.

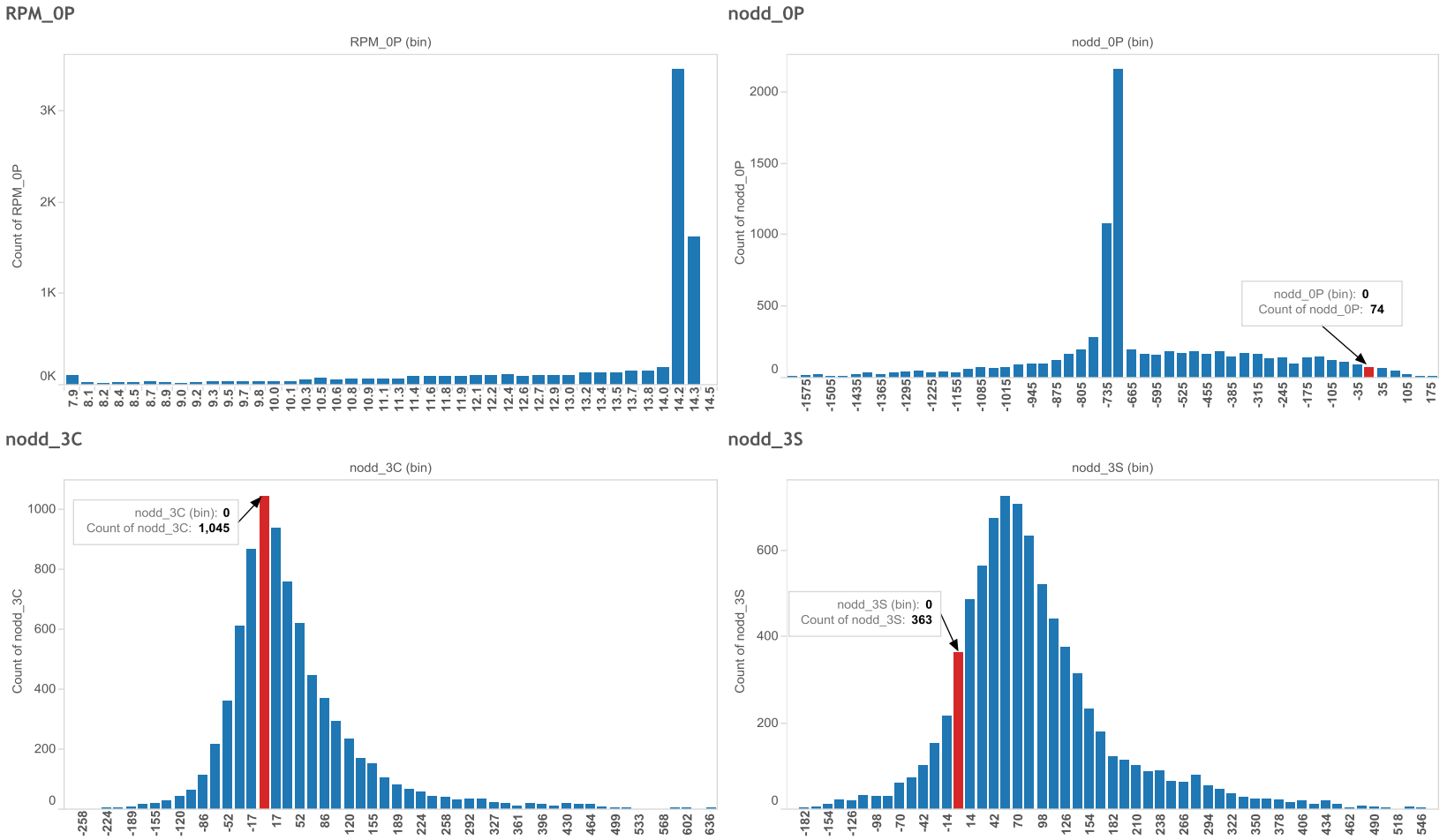
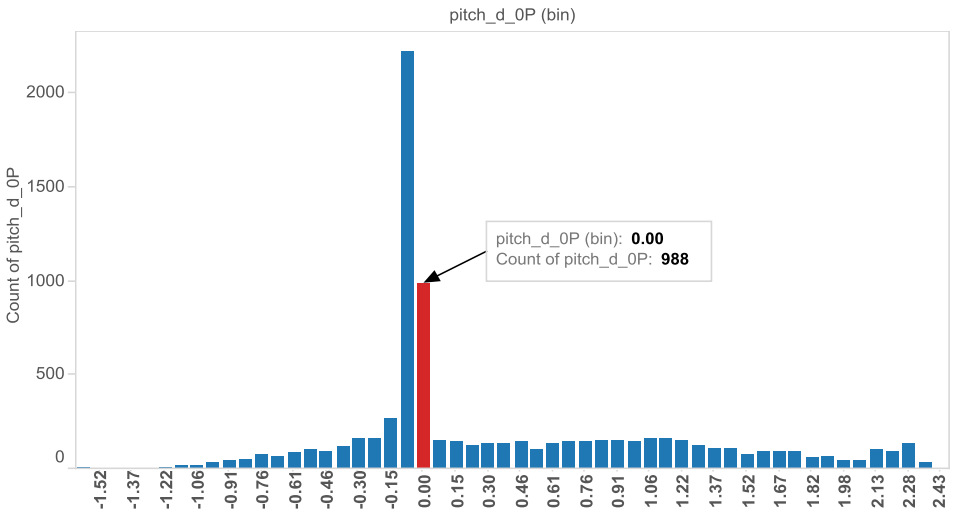
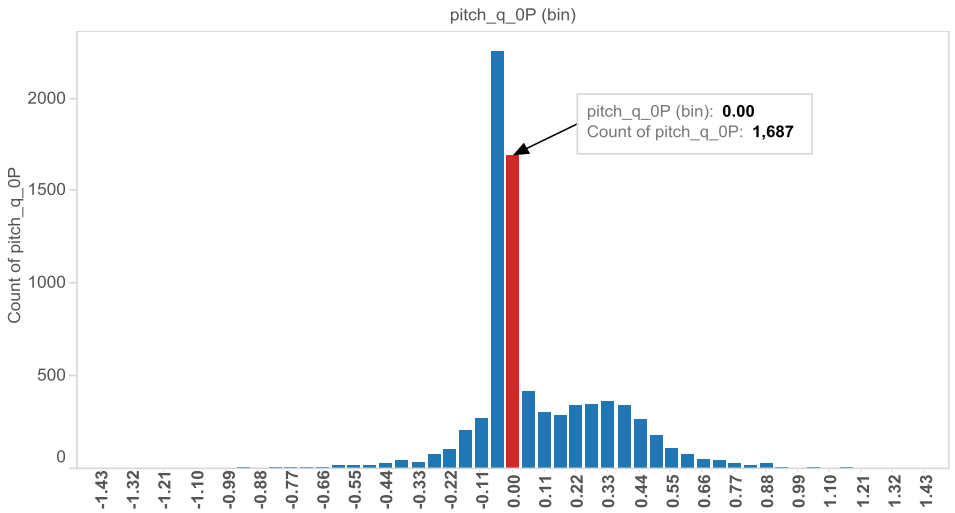


FIGURE 4 HISTOGRAMS FOR RPM_0P, NODD_0P, NODD_3C AND NODD_3S

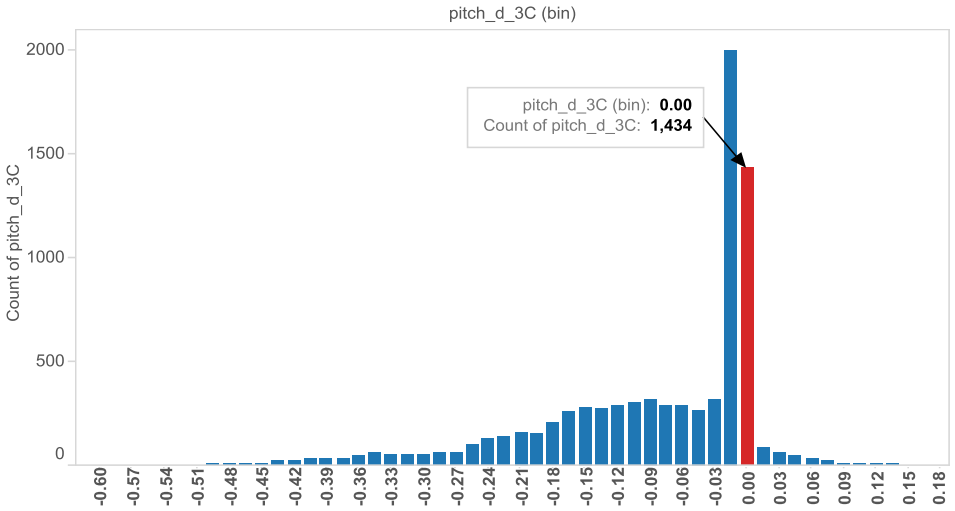
pitch_d_0P



pitch_q_0P



pitch_d_3C



pitch_d_3S

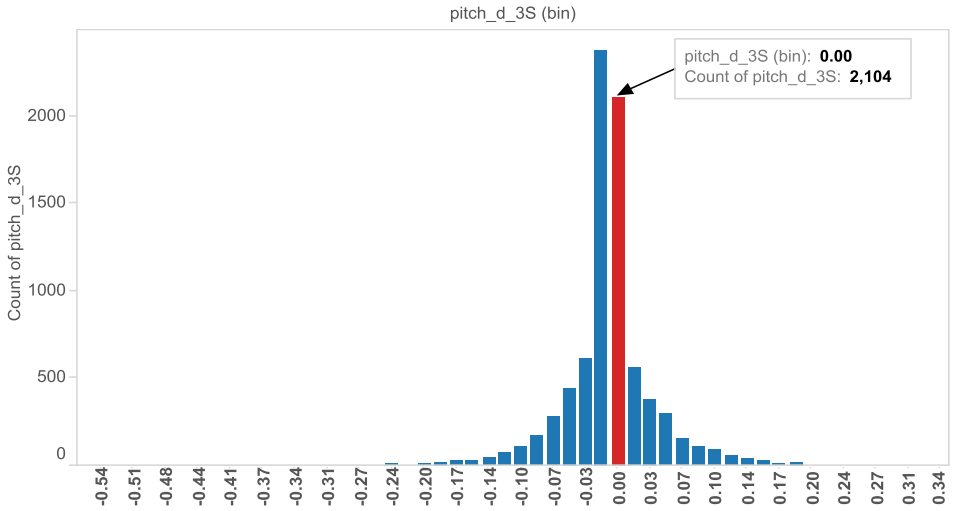


FIGURE 5 HISTOGRAMS FOR PITCH_D_0P, PITCH_Q_0P, PITCH_D_3C AND PITCH_D_3S

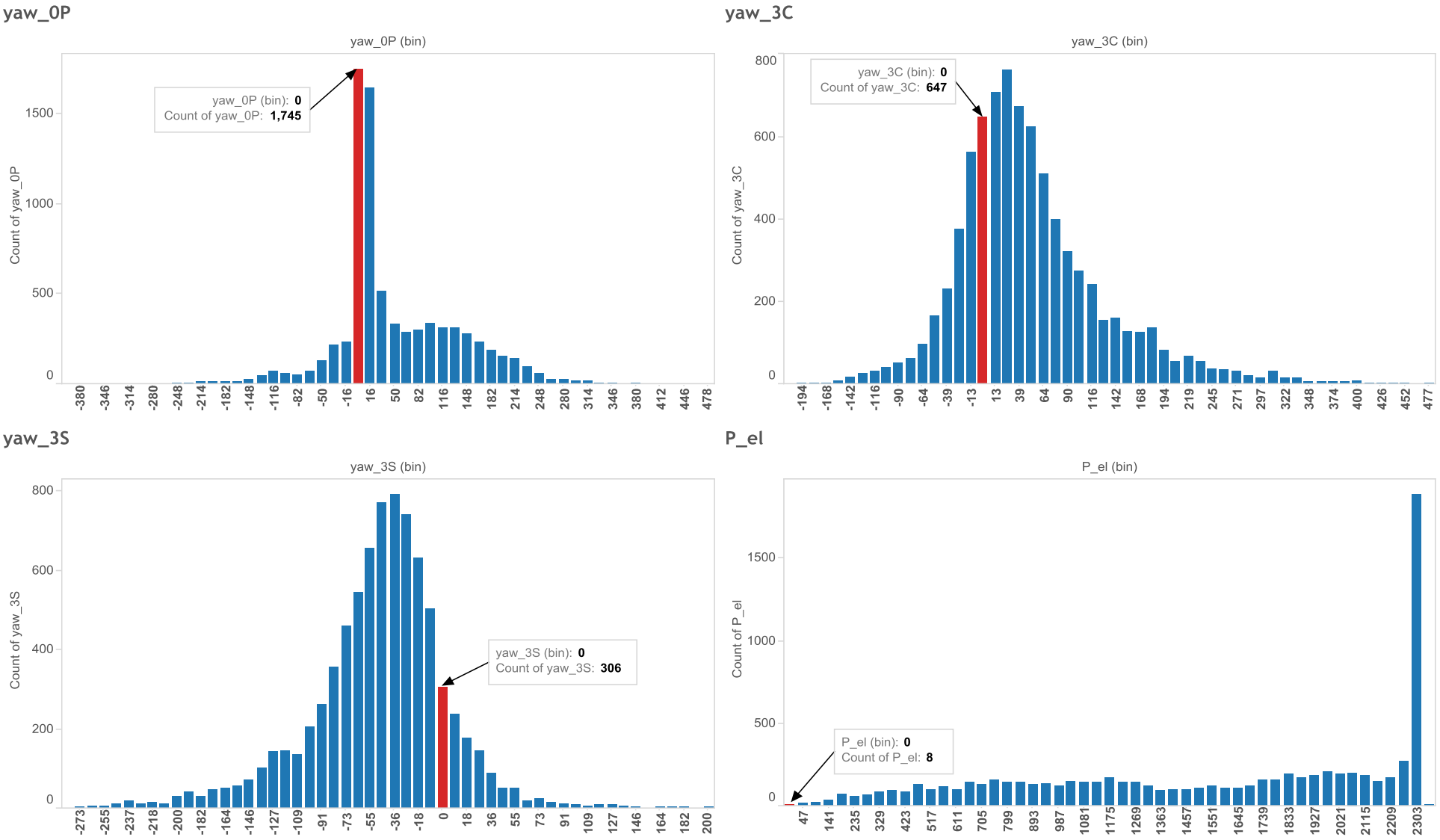
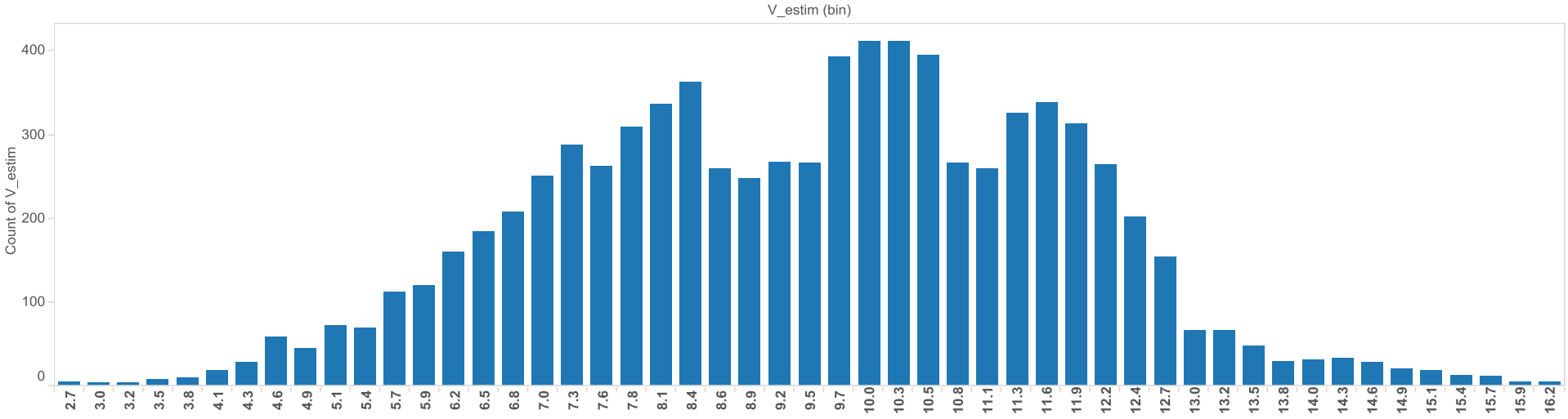


FIGURE 6 HISTOGRAMS FOR YAW_0P, YAW_3C, YAW_3S AND P_EL

V_estim



pitch_col_0P

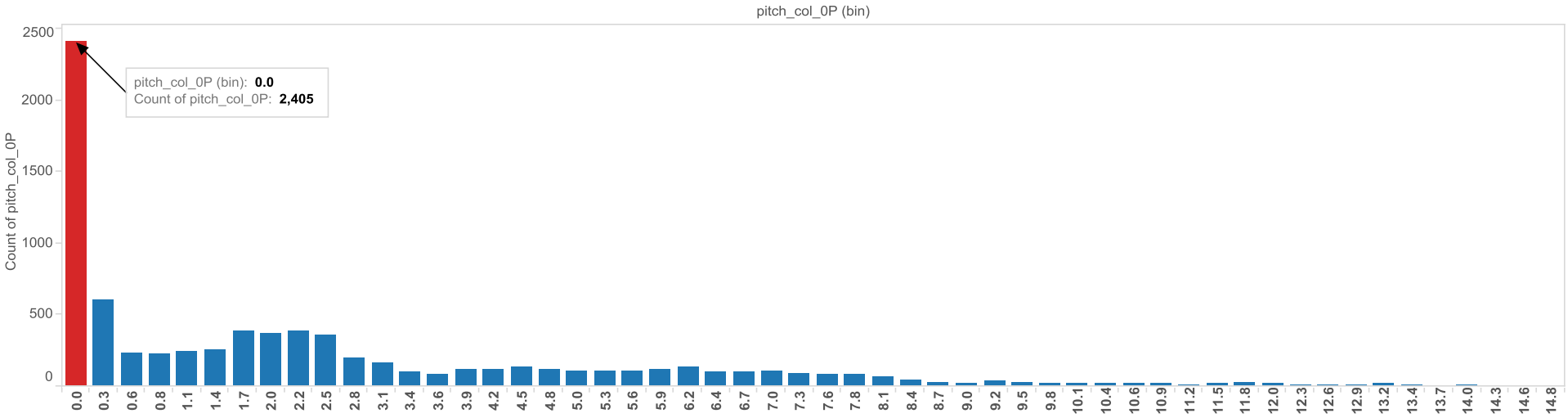


FIGURE 7 HISTOGRAMS FOR V_ESTIM AND PITCH_COL_0P

The following charts show how each sensor data is distributed according to its Shear Type. Depending on Shear Type, there are some cases where data looks symmetry, and for some sensor data variables its symmetry is affected when data is mix between Shear Types.

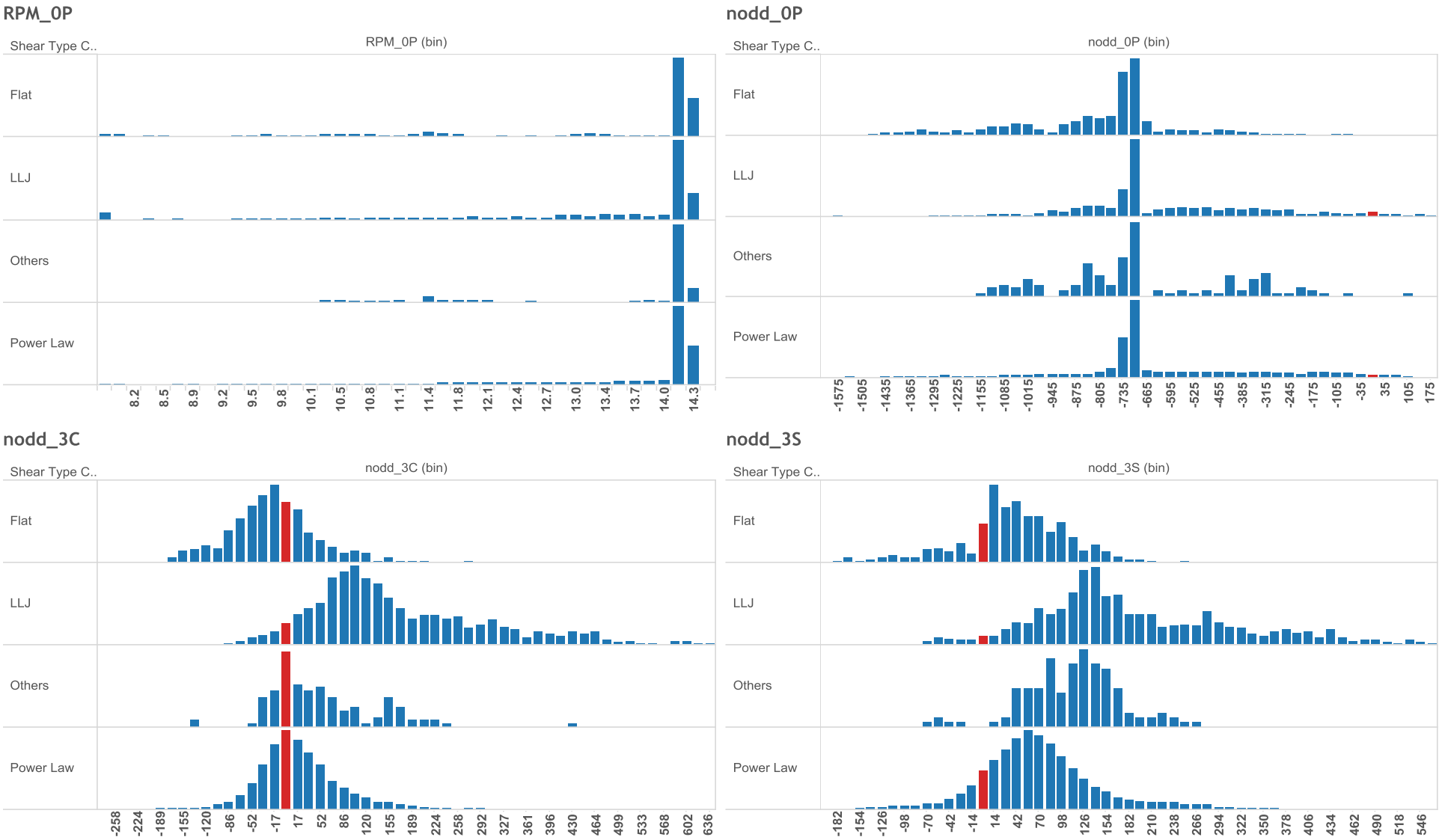


FIGURE 8 HISTOGRAMS BY SHEAR TYPE FOR RPM_0P, NODD_0P, NODD_3C AND NODD_3S

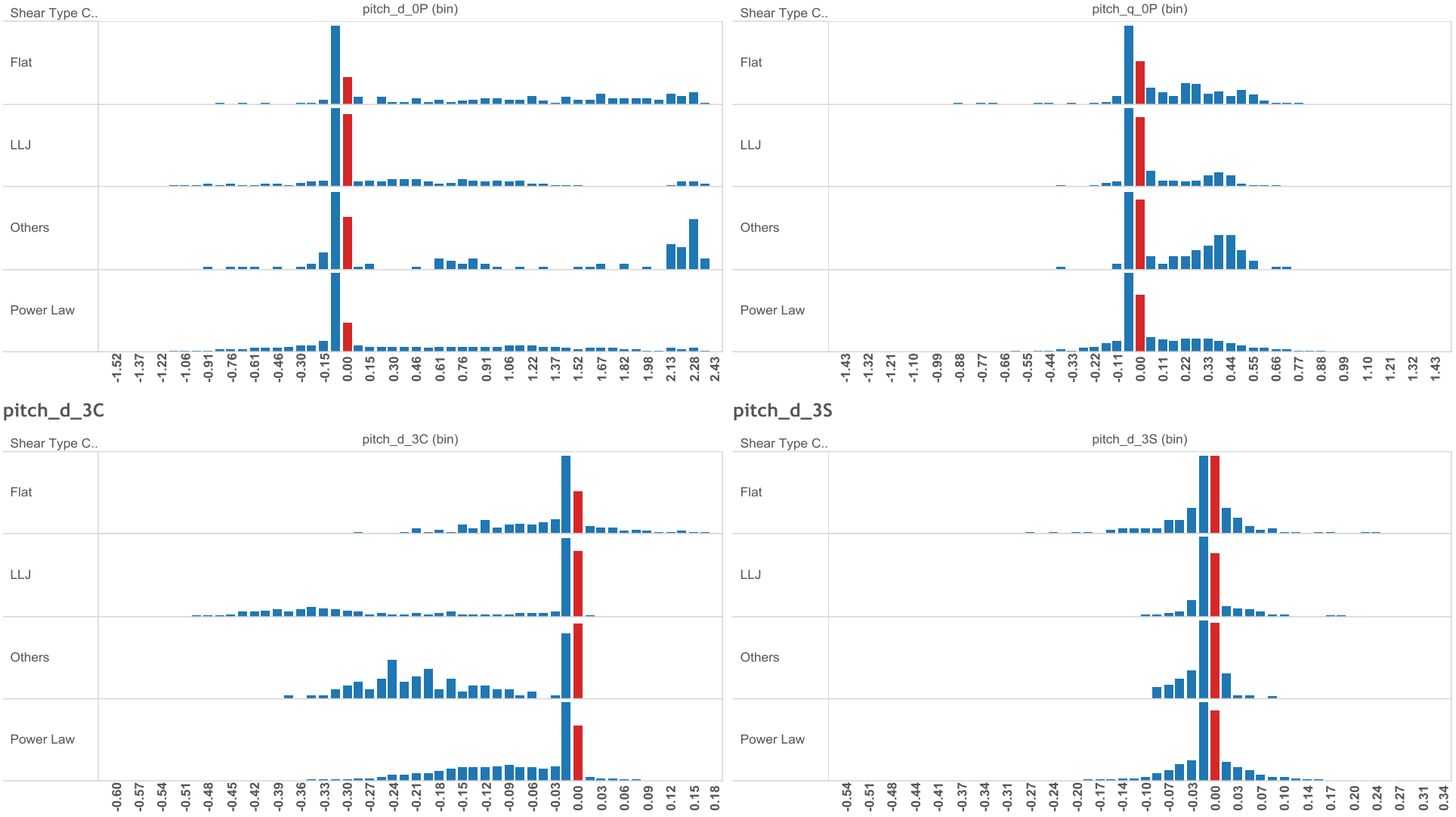


FIGURE 9 HISTOGRAMS BY SHEAR TYPE FOR PITCH_D_0P, PITCH_Q_0P, PITCH_D_3C AND PITCH_D_3S

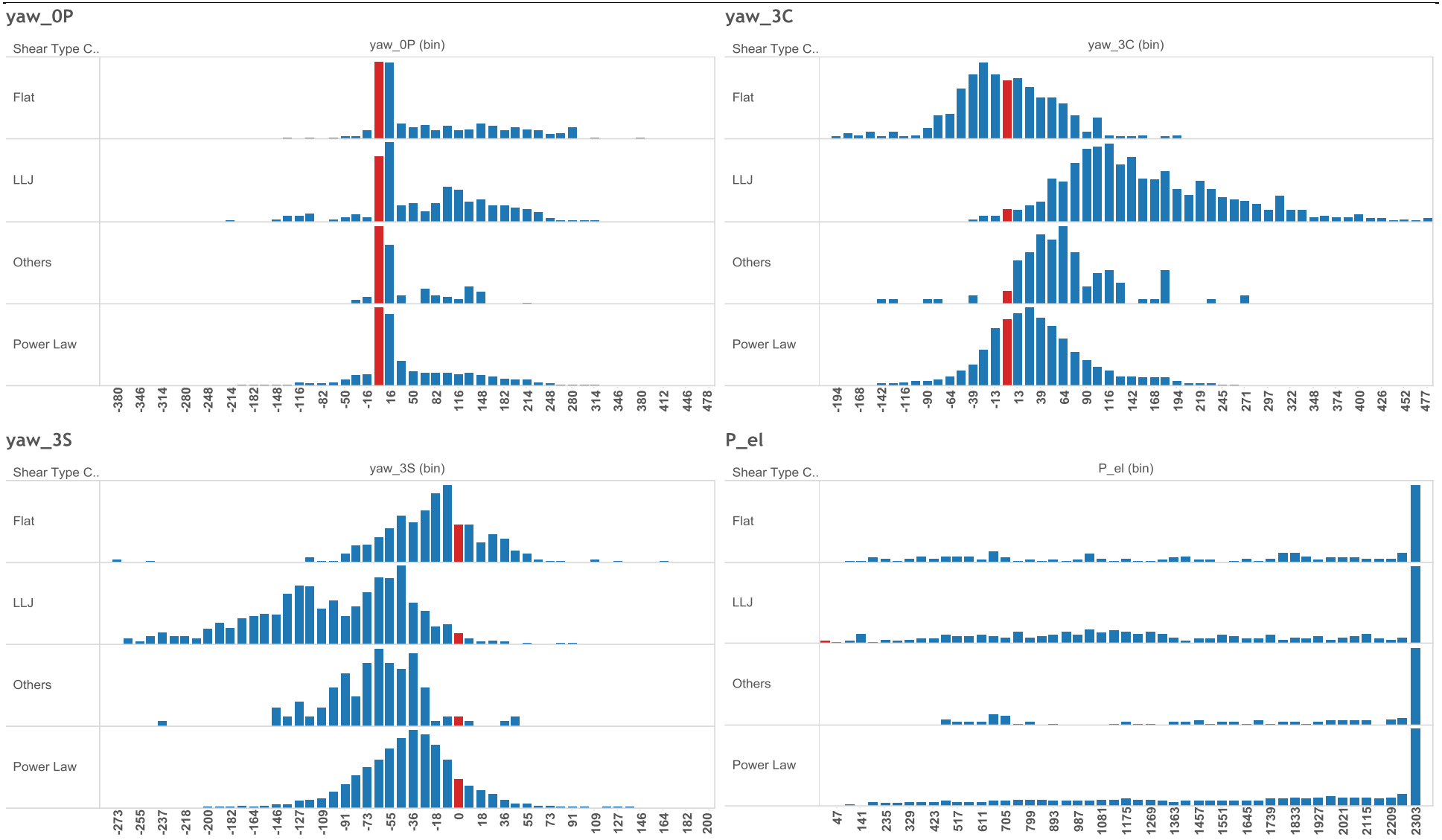


FIGURE 10 HISTOGRAMS BY SHEAR TYPE FOR YAW_0P, YAW_3C, YAW_3S AND P_EL

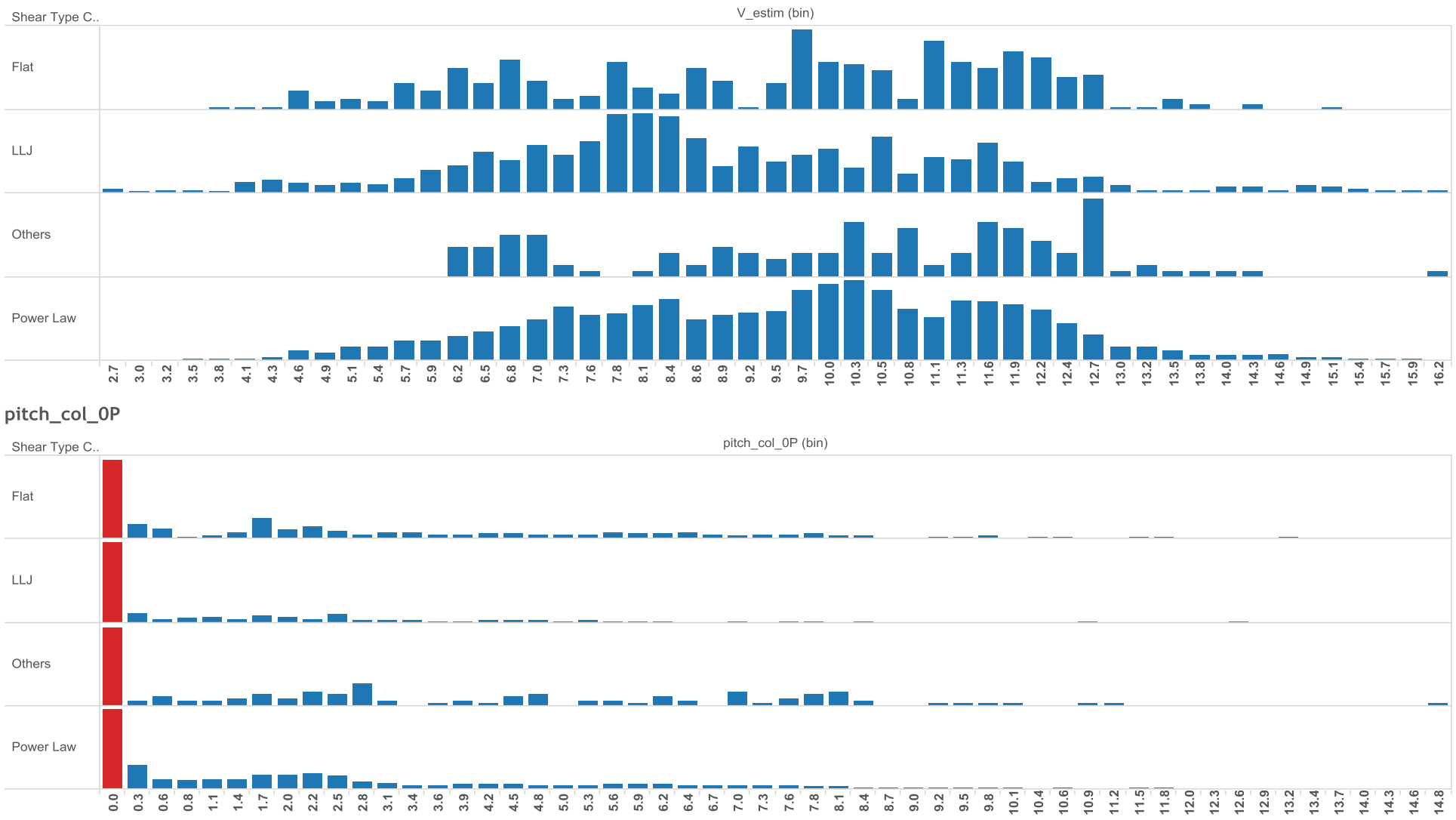


FIGURE 11 HISTOGRAMS BY SHEAR TYPE FOR V_ESTIM AND PITCH_COL_0P

Let's analyze each sensor data variable per day and see how data fits into its **± 3 standard deviations**. ±3 standard deviations are calculated using all data for that specific variable. See how many data is outside of its ±3 standard deviations (some are highlighted with a red square). ***So, it's recommended to normalize data or remove outliers in order to select which statistic model has better predictive performance***. See below charts.

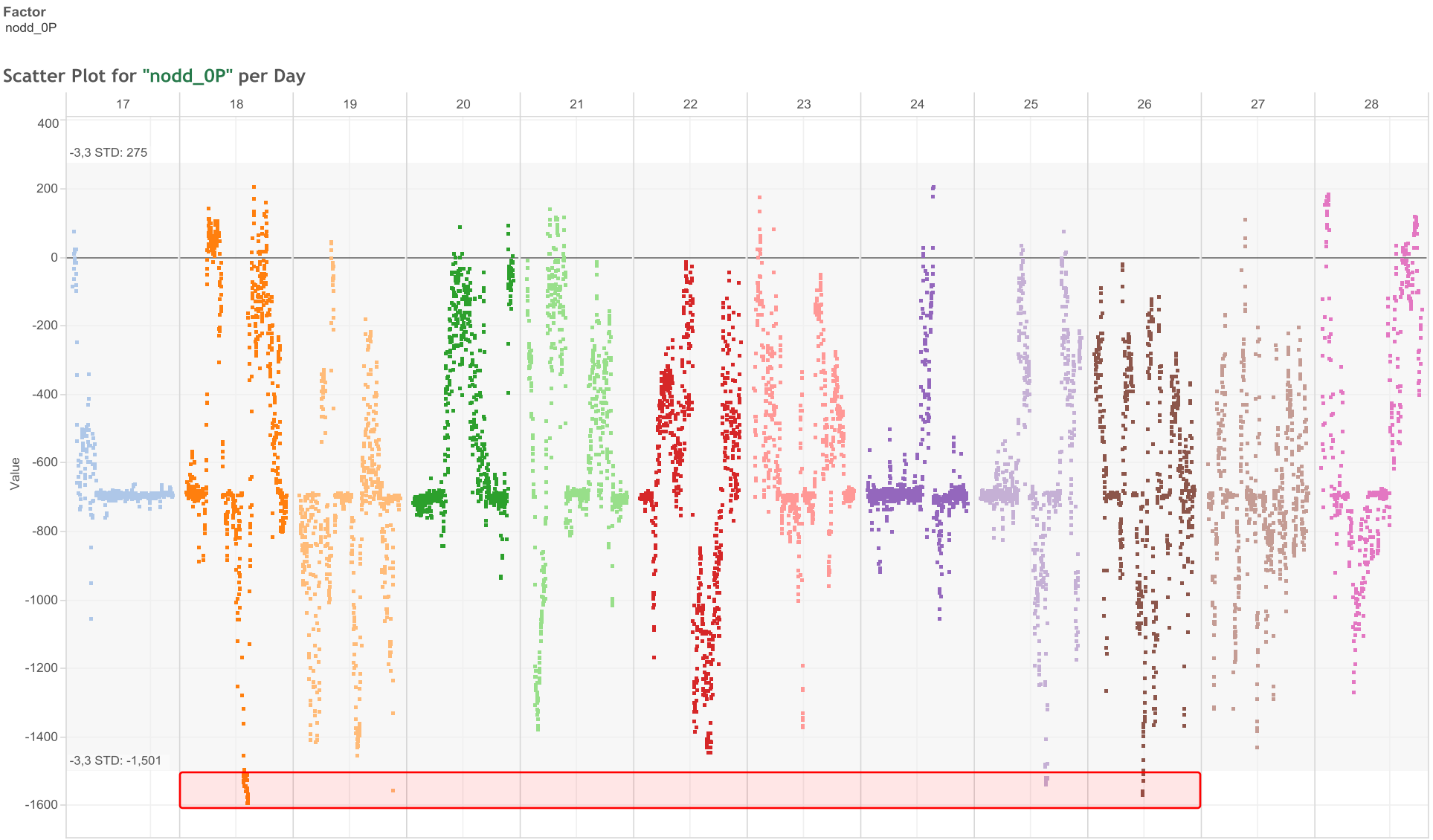


FIGURE 12 SCATTER PLOT FOR NODD_0P BY DAY VS. ± 3 STANDARD DEVIATION.

Factor
nodd_3C

Scatter Plot for "nodd_3C" per Day

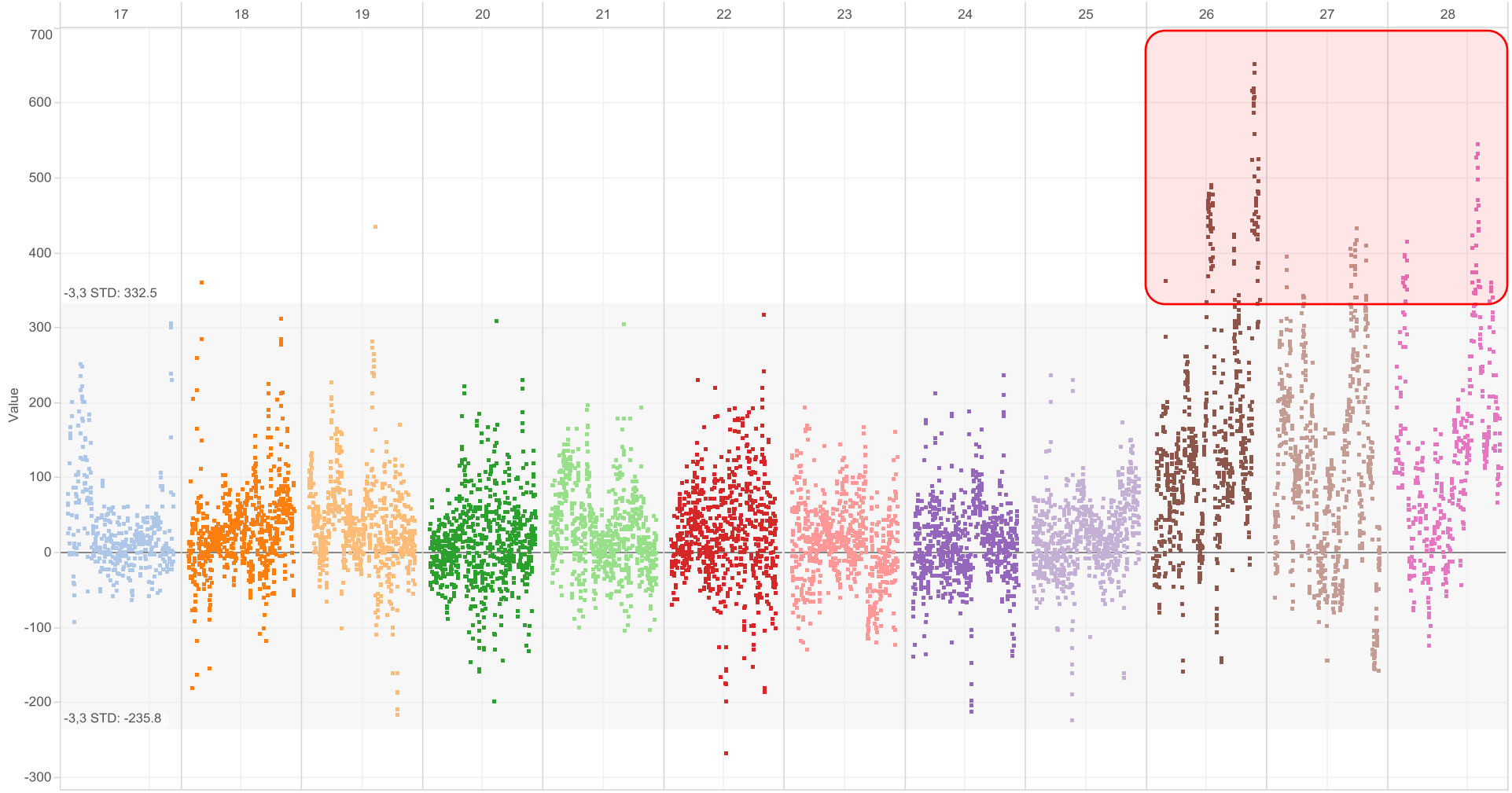


FIGURE 13 SCATTER PLOT FOR NODD_3C BY DAY VS. ± 3 STANDARD DEVIATION.

Factor
nodd_3S

Scatter Plot for "nodd_3S" per Day

FIGURE 14 SCATTER PLOT FOR NODD_3S BY DAY VS. ± 3 STANDARD DEVIATION.

Factor
P_el

Scatter Plot for "P_el" per Day

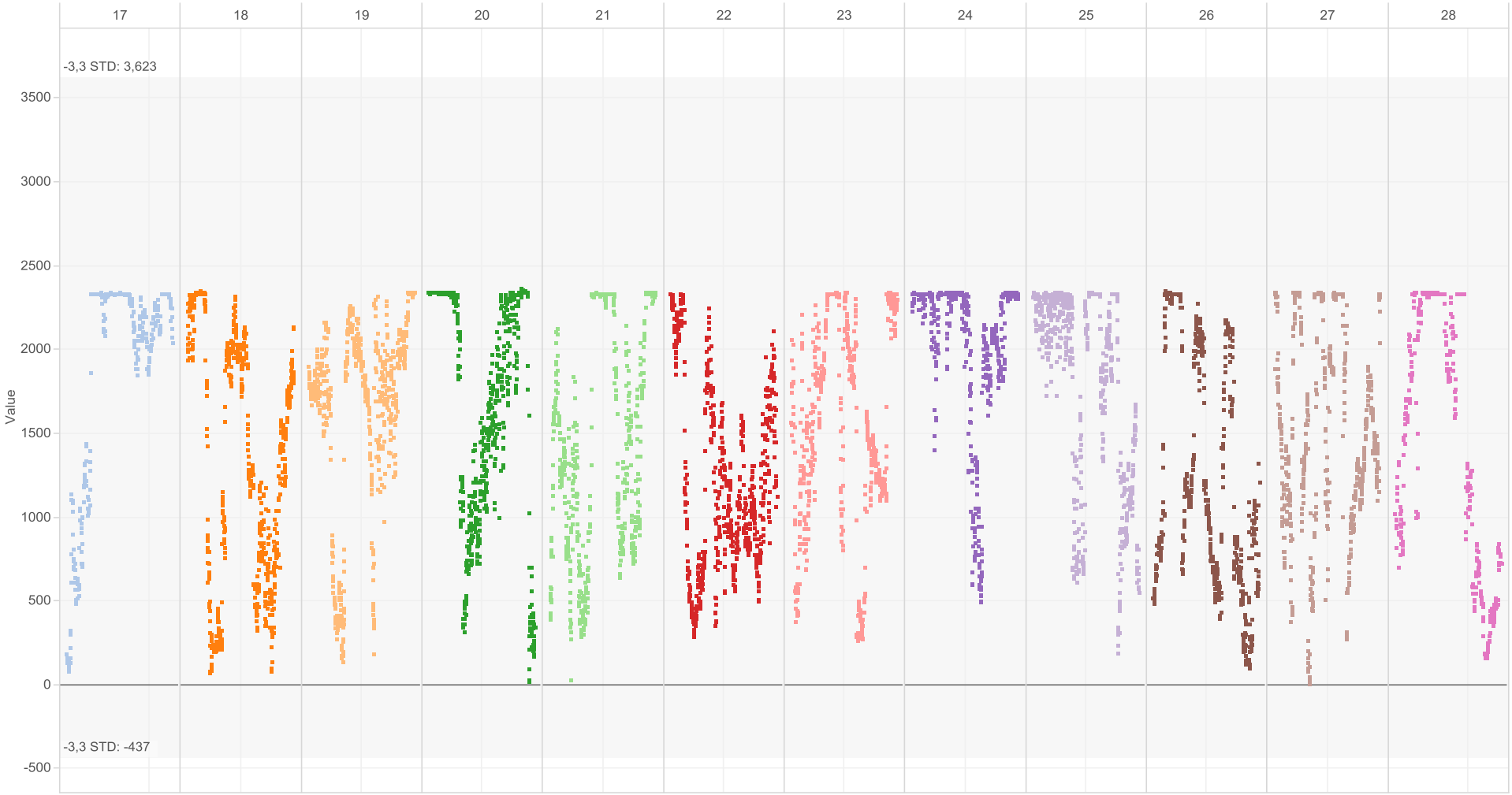


FIGURE 15 SCATTER PLOT FOR P_EL BY DAY VS. ± 3 STANDARD DEVIATION.

Factor
pitch_col_0P

Scatter Plot for "pitch_col_0P" per Day

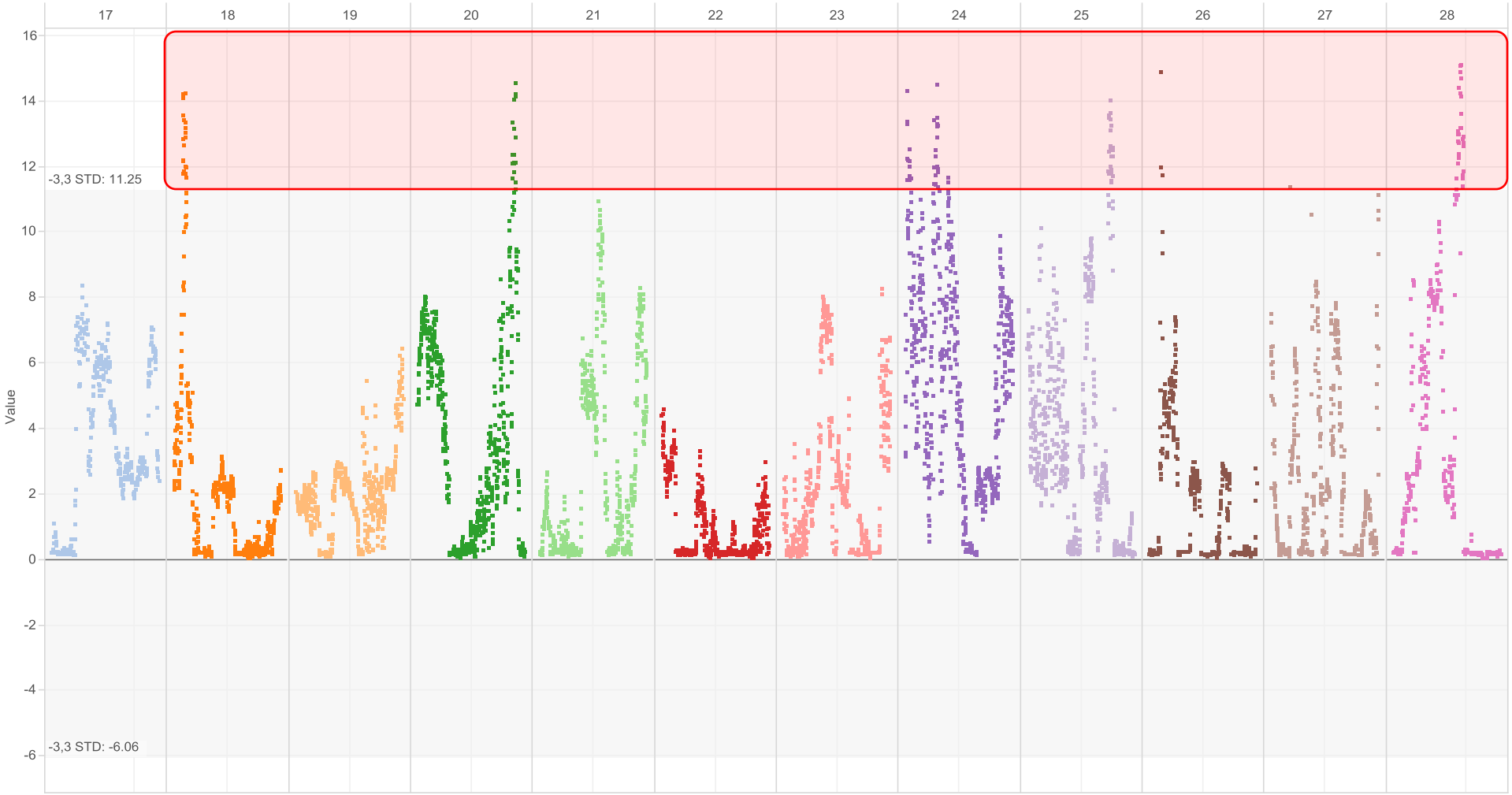


FIGURE 16 SCATTER PLOT FOR PITCH_COL_0P BY DAY VS. ± 3 STANDARD DEVIATION.

Factor
pitch_d_0P

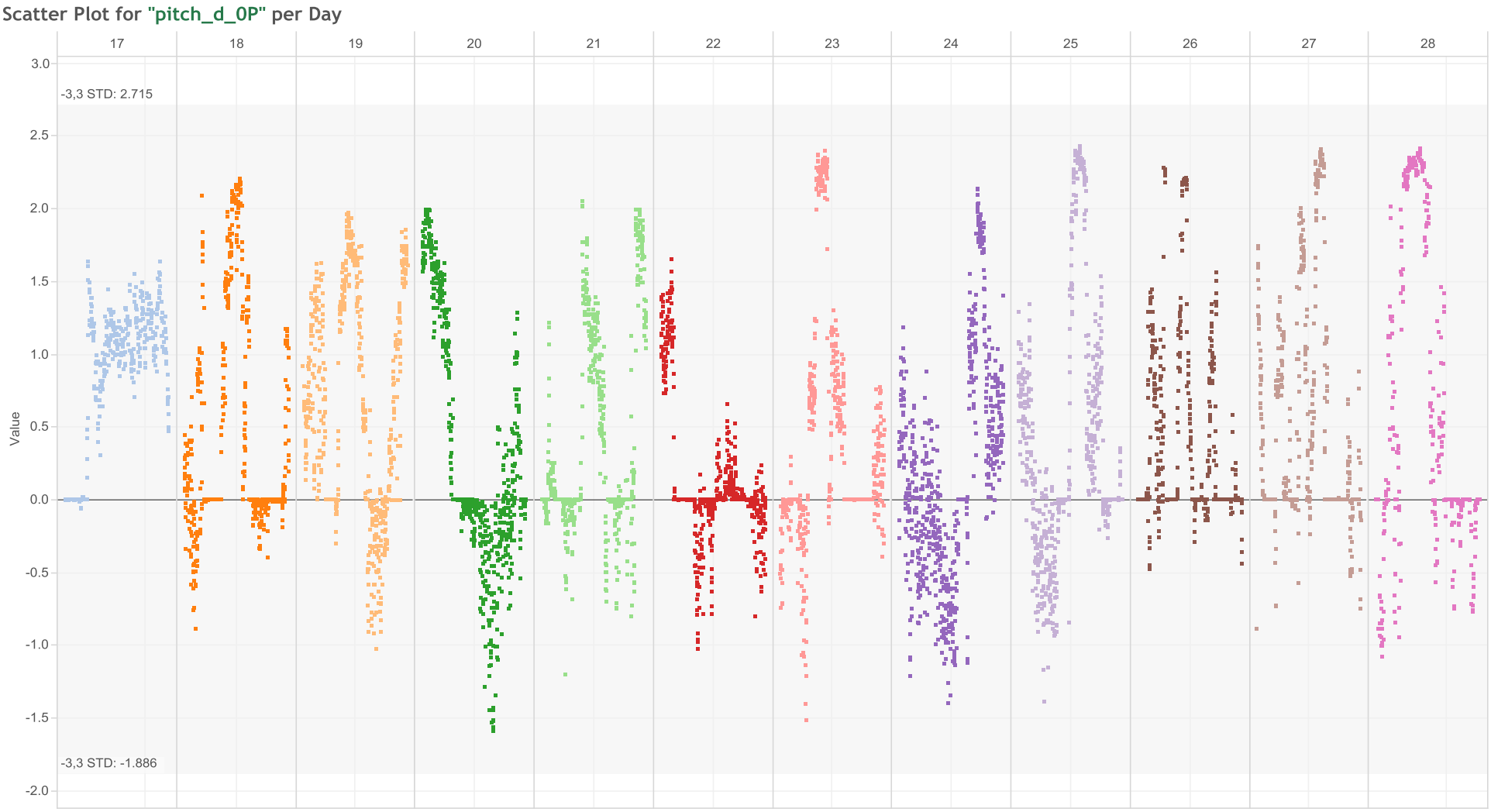


FIGURE 17 SCATTER PLOT FOR PITCH_D_0P BY DAY VS. ± 3 STANDARD DEVIATION.

Factor
pitch_d_3C

Scatter Plot for "pitch_d_3C" per Day

FIGURE 18 SCATTER PLOT FOR PITCH_D_3C BY DAY VS. ± 3 STANDARD DEVIATION.

Factor
pitch_d_3S

Scatter Plot for "pitch_d_3S" per Day

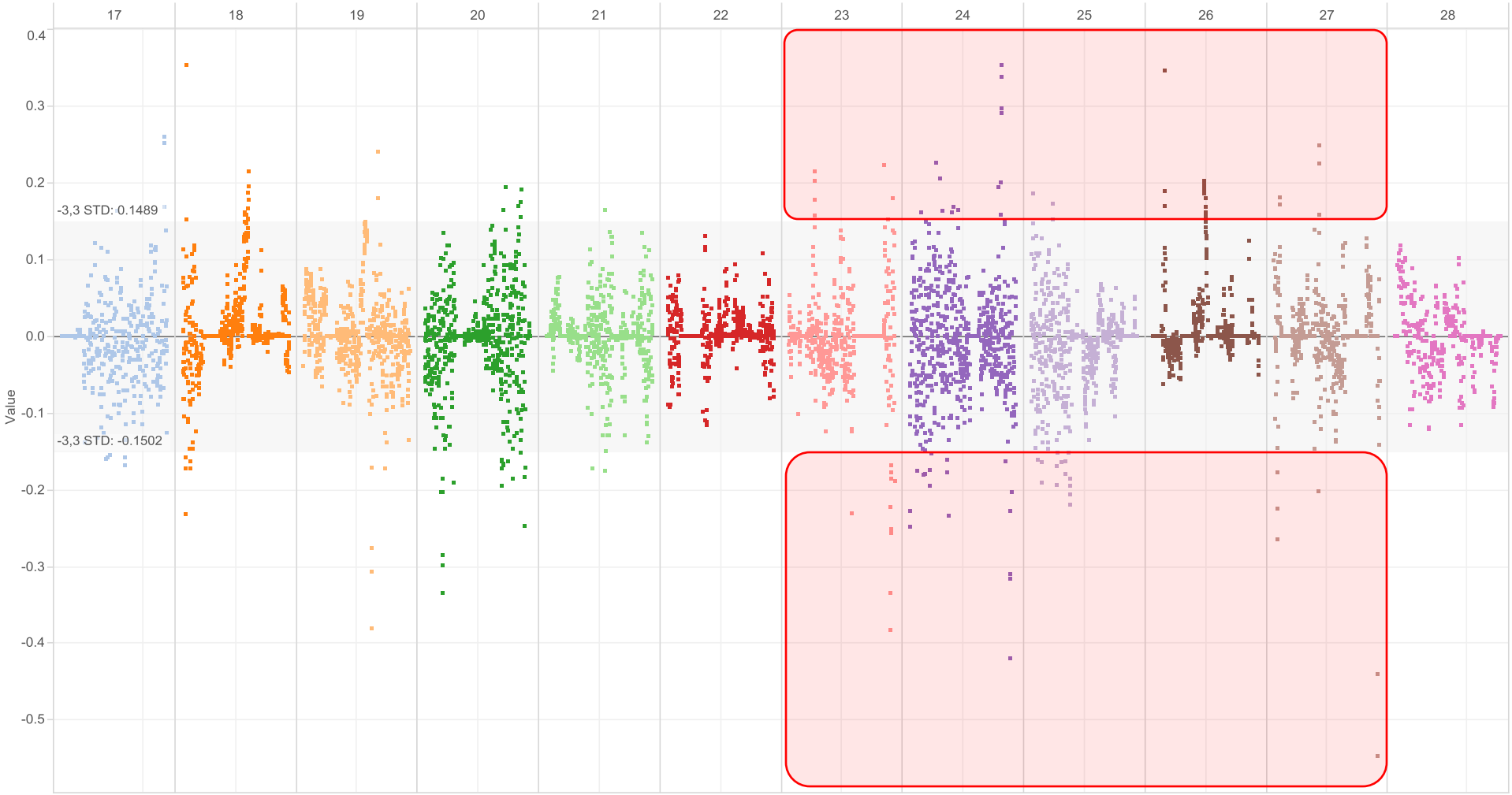


FIGURE 19 SCATTER PLOT FOR PITCH_D_3S BY DAY VS. ± 3 STANDARD DEVIATION.

Factor
pitch_q_0P

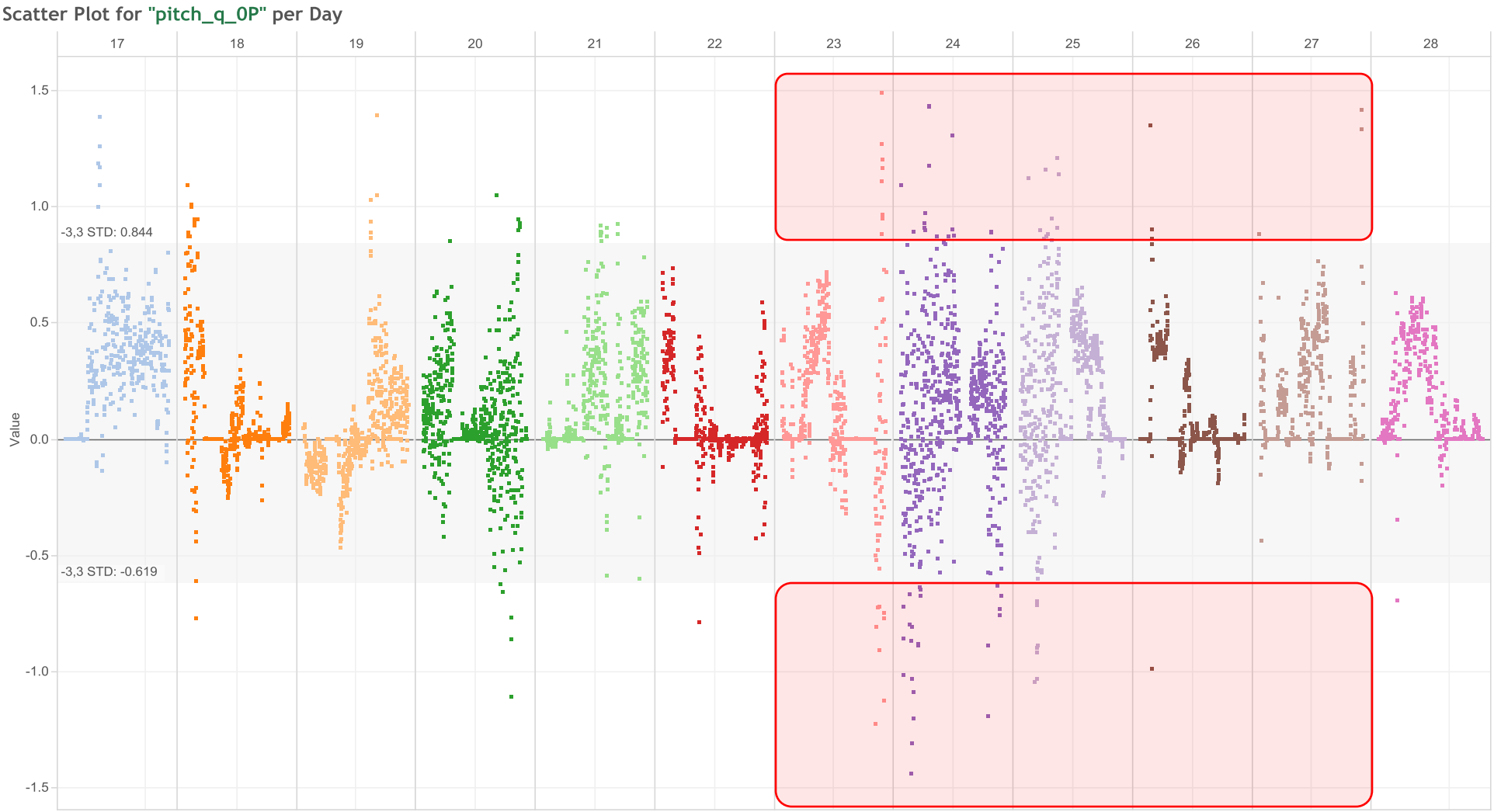


FIGURE 20 SCATTER PLOT FOR PITCH_Q_0P BY DAY VS. ± 3 STANDARD DEVIATION.

Factor
RPM_0P

Scatter Plot for "RPM_0P" per Day

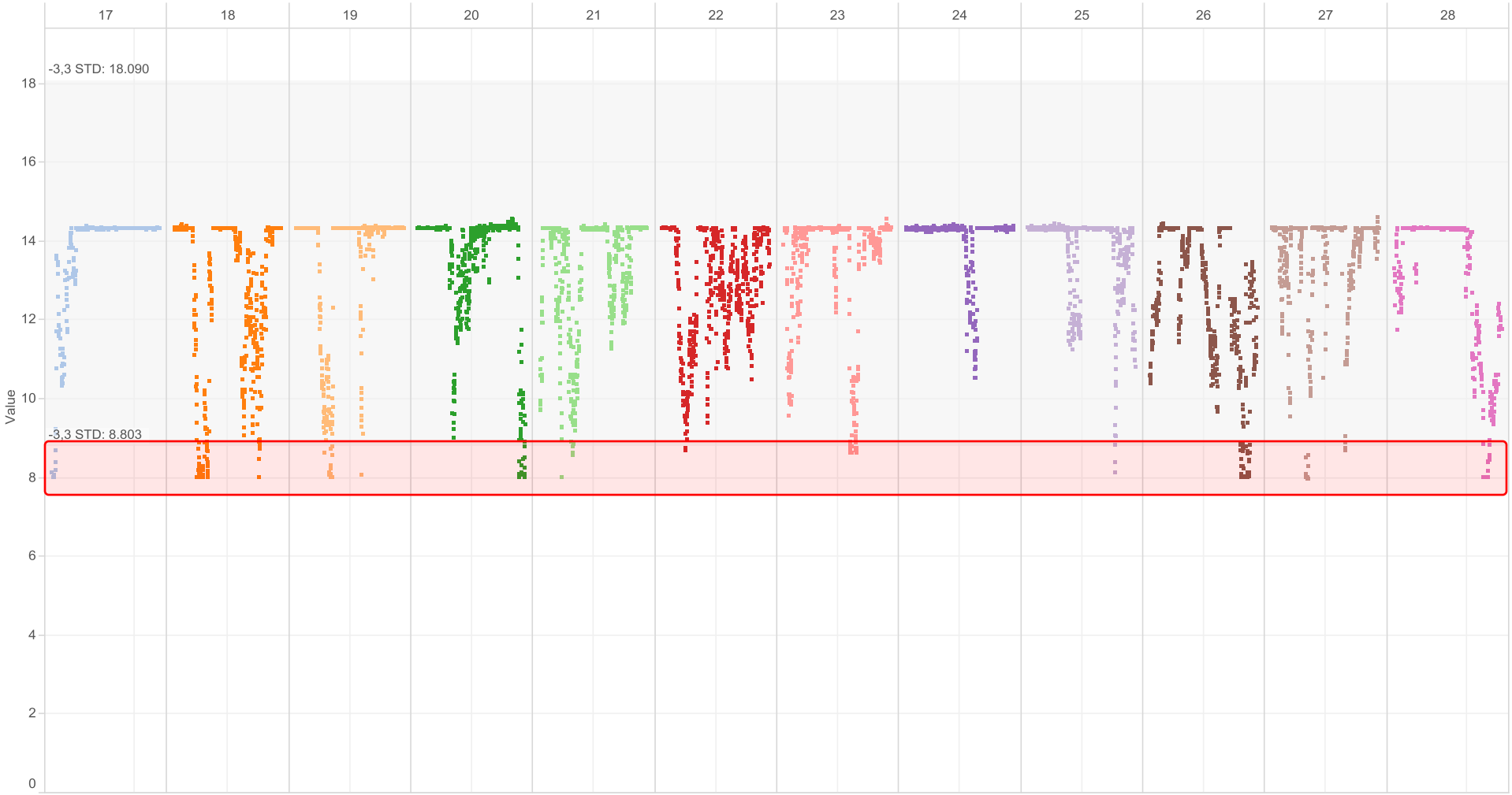


FIGURE 21 SCATTER PLOT FOR RPM_0P BY DAY VS. ± 3 STANDARD DEVIATION.

Factor
V_estim

Scatter Plot for "V_estim" per Day

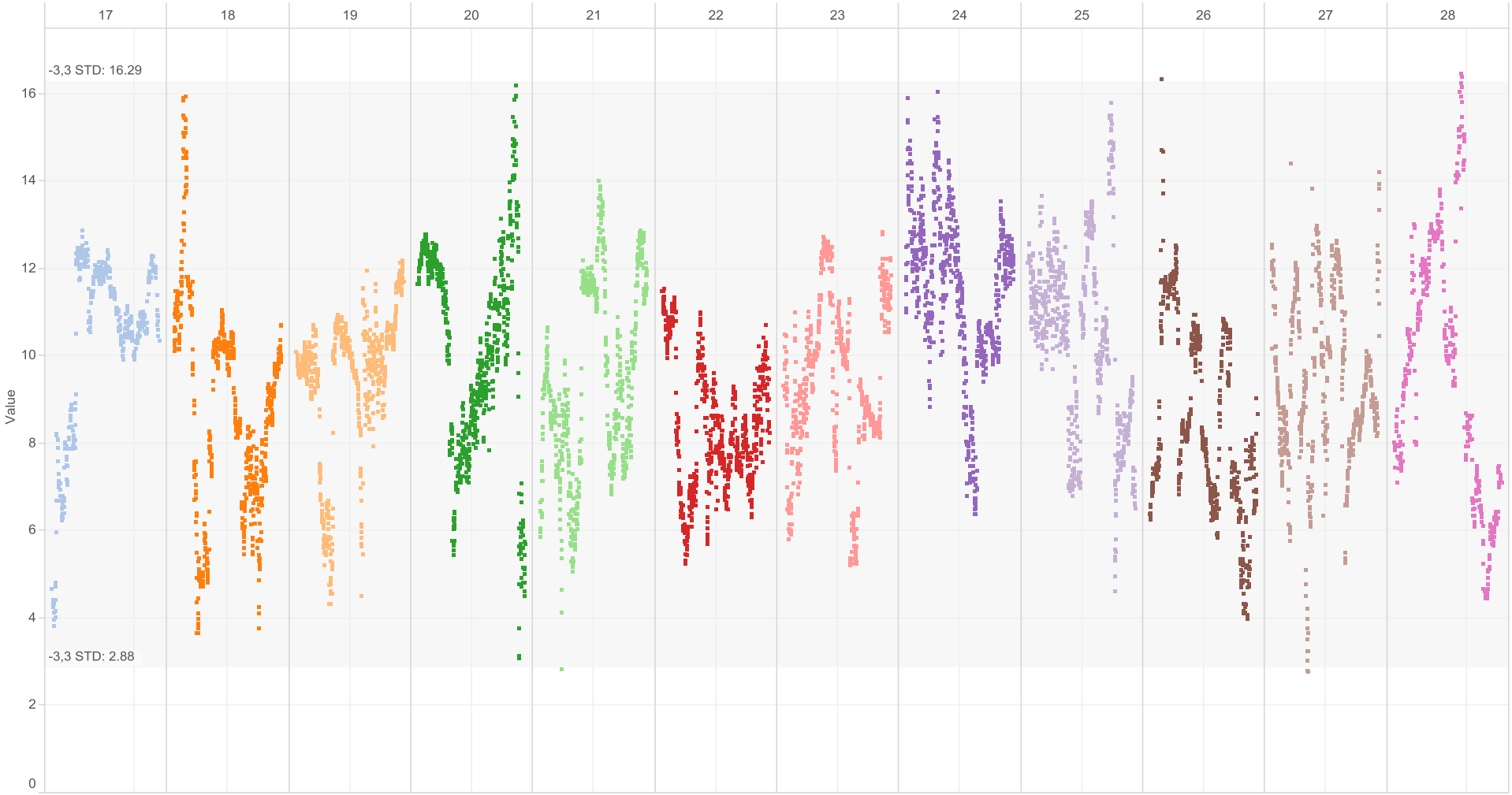


FIGURE 22 SCATTER PLOT FOR V_ESTIM BY DAY VS. ± 3 STANDARD DEVIATION.

Factor
yaw_0P

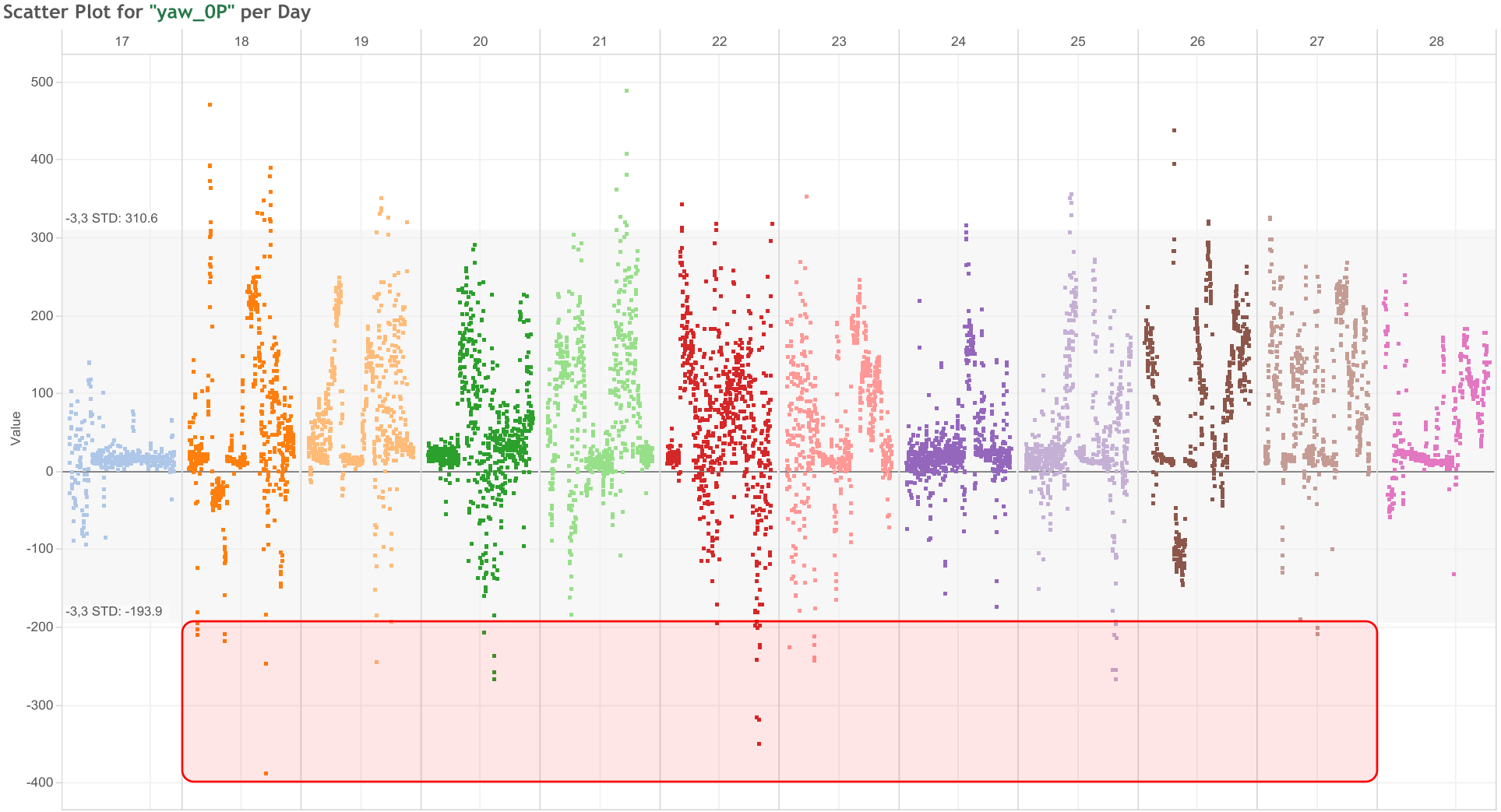


FIGURE 23 SCATTER PLOT FOR YAW_0P BY DAY VS. ± 3 STANDARD DEVIATION.

Factor
yaw_3C

Scatter Plot for "yaw_3C" per Day

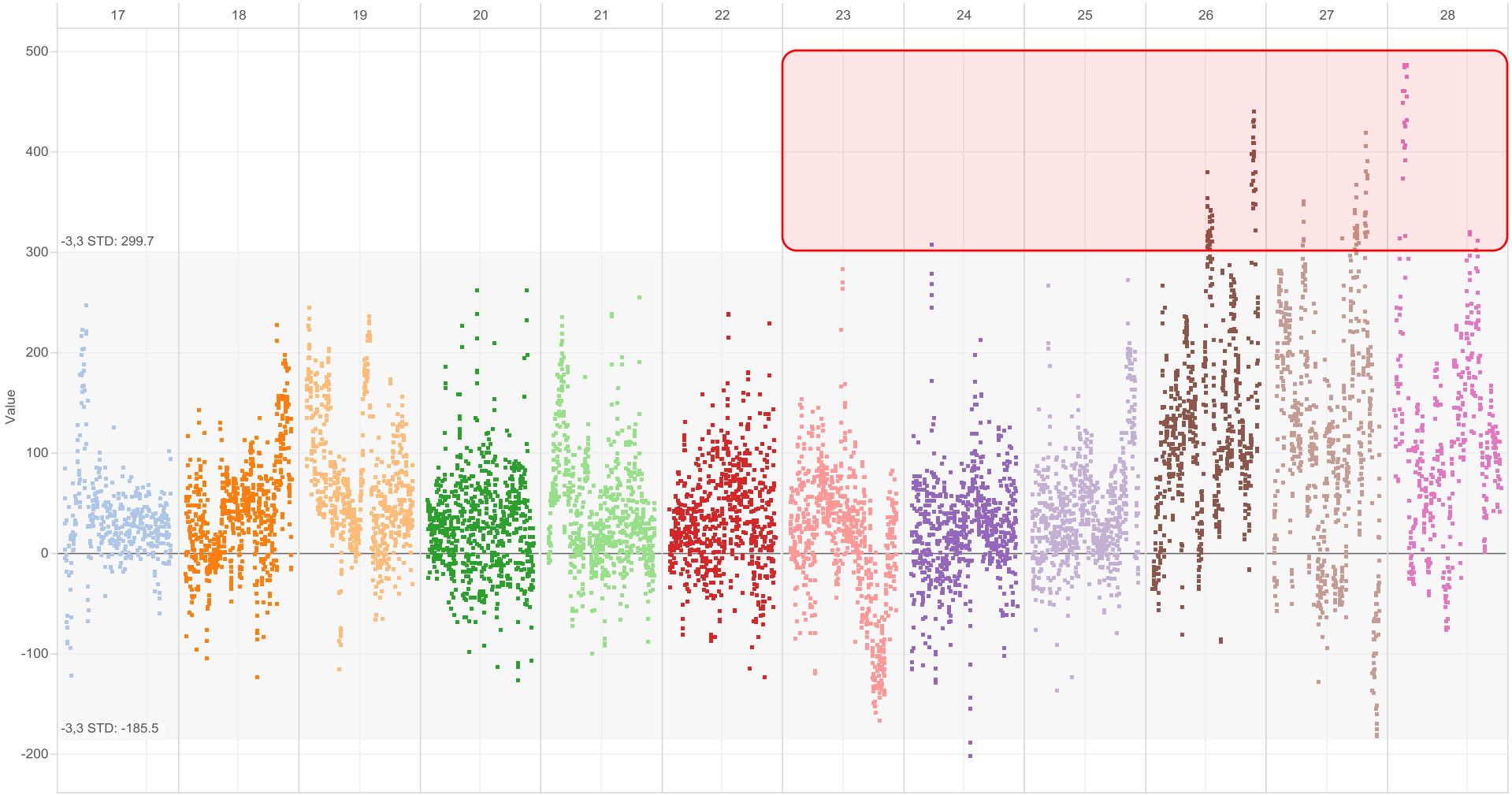


FIGURE 24 SCATTER PLOT FOR YAW_3C BY DAY VS. ± 3 STANDARD DEVIATION.

Factor
yaw_3S

Scatter Plot for "yaw_3S" per Day

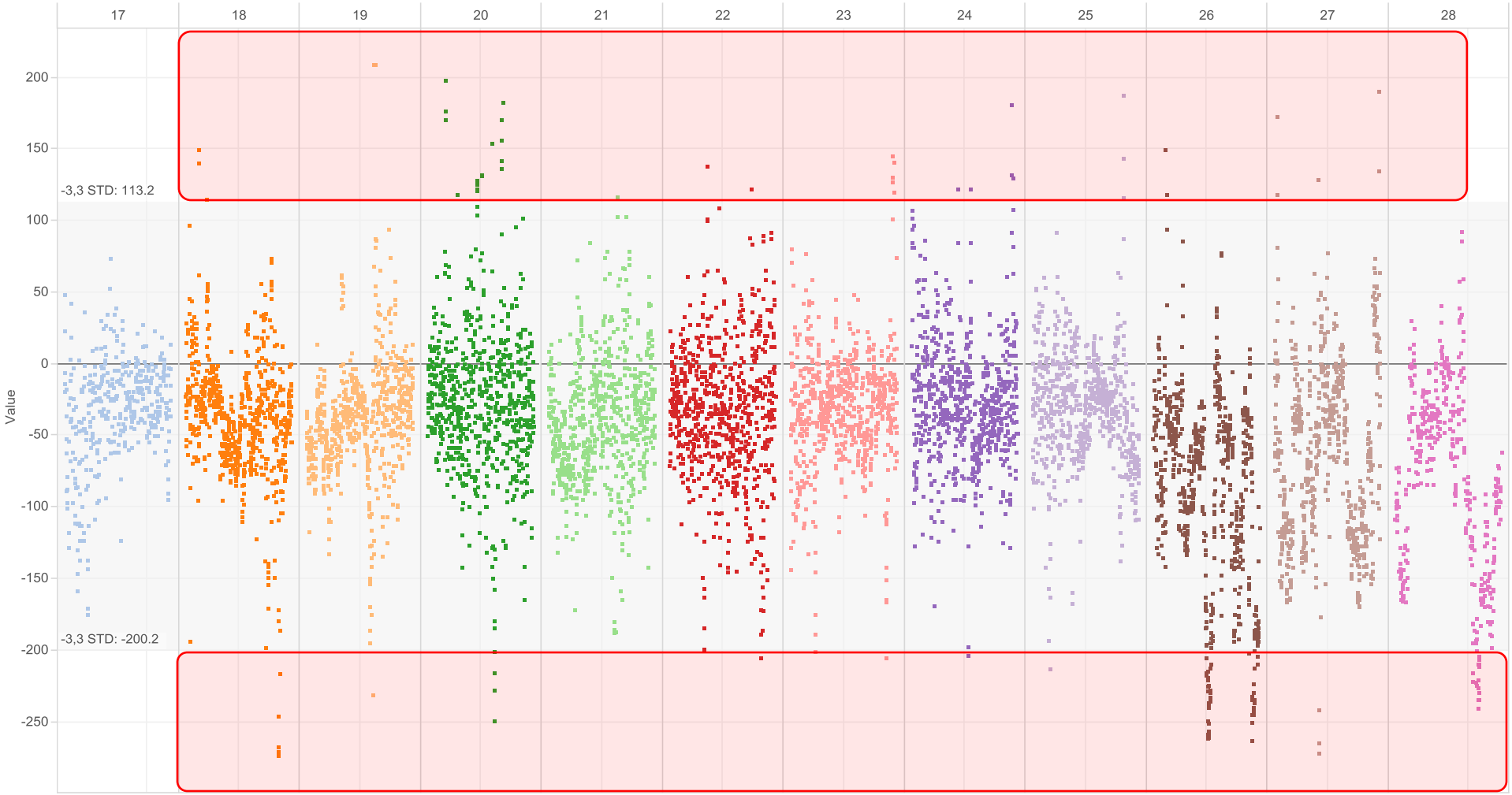


FIGURE 25 SCATTER PLOT FOR YAW_3S BY DAY VS. ± 3 STANDARD DEVIATION.

All these scatter plots can be complemented analyzing each shear type using box-plot, in order to see its data distribution per day, check the minimum, first quartile, median, third quartile, maximum and outliers. See a couple of charts below. Further details can be found into ***TABLEU Packaged Workbook*** ("Wind Shear Analysis.twbx" – included in the deliverables).

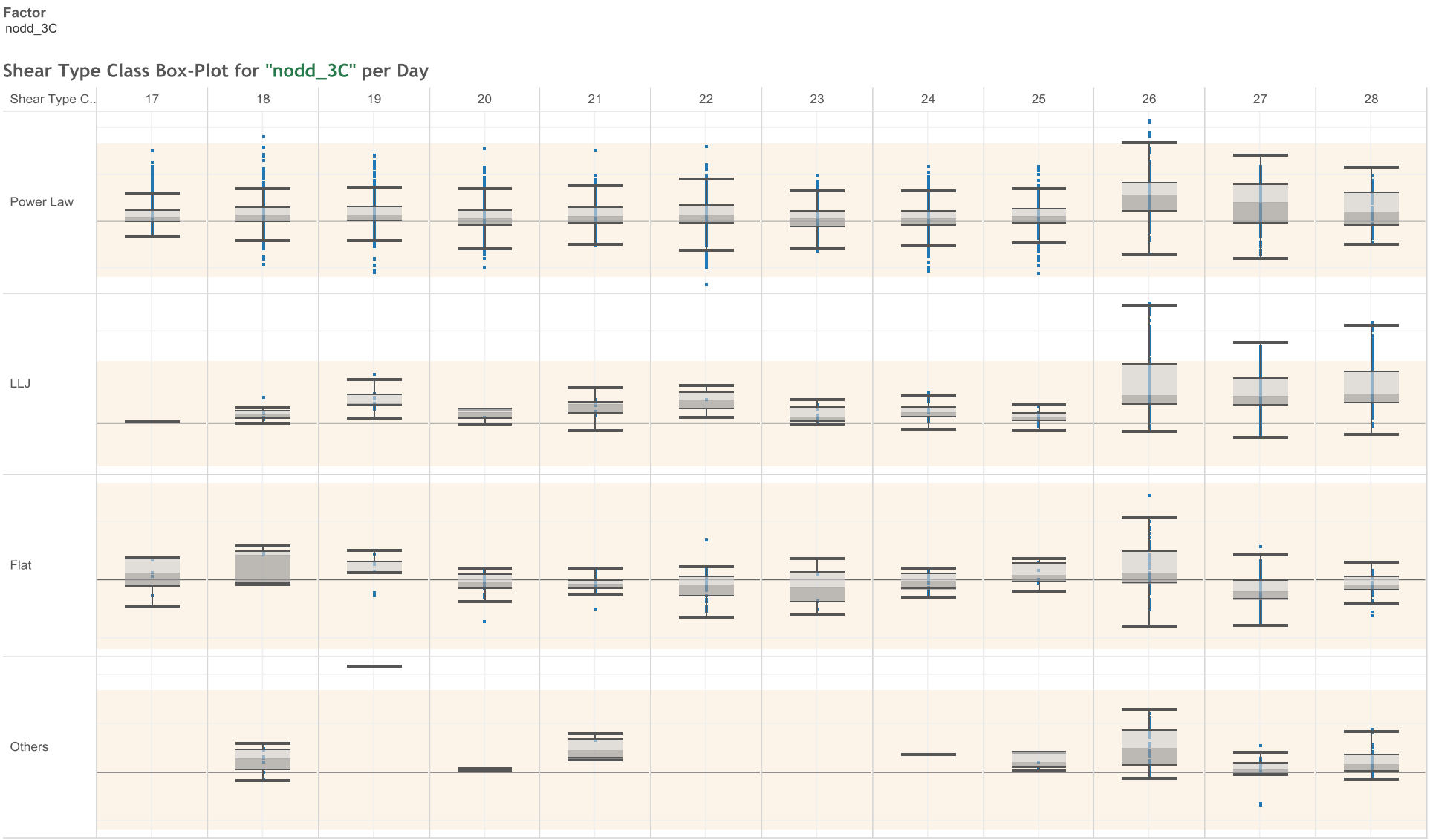


FIGURE 26 BOX PLOT FOR NODD_3C BY DAY.

Factor
pitch_d_3S

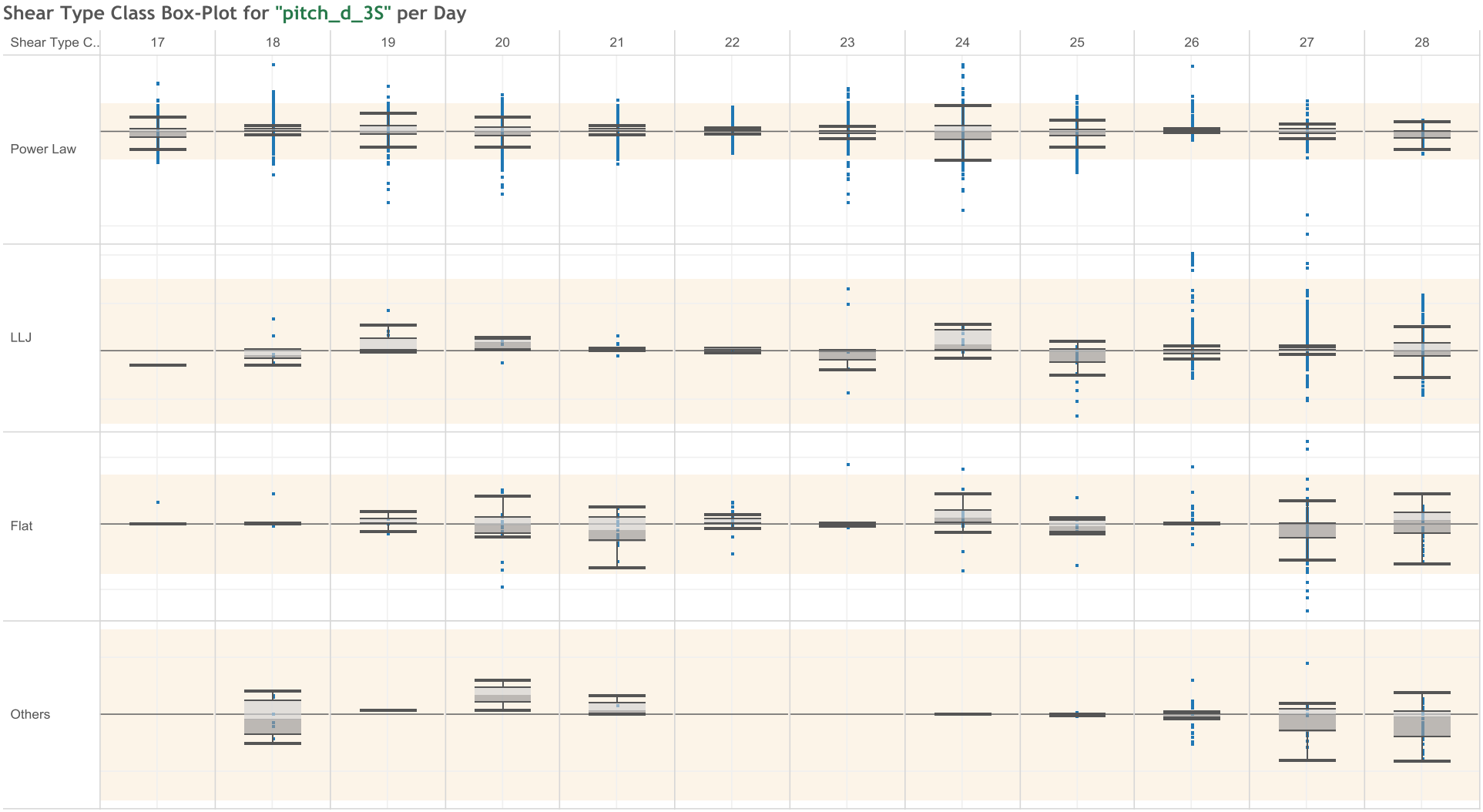


FIGURE 27 BOX PLOT FOR PITCH_D_3S BY DAY.

Factor
yaw_3S

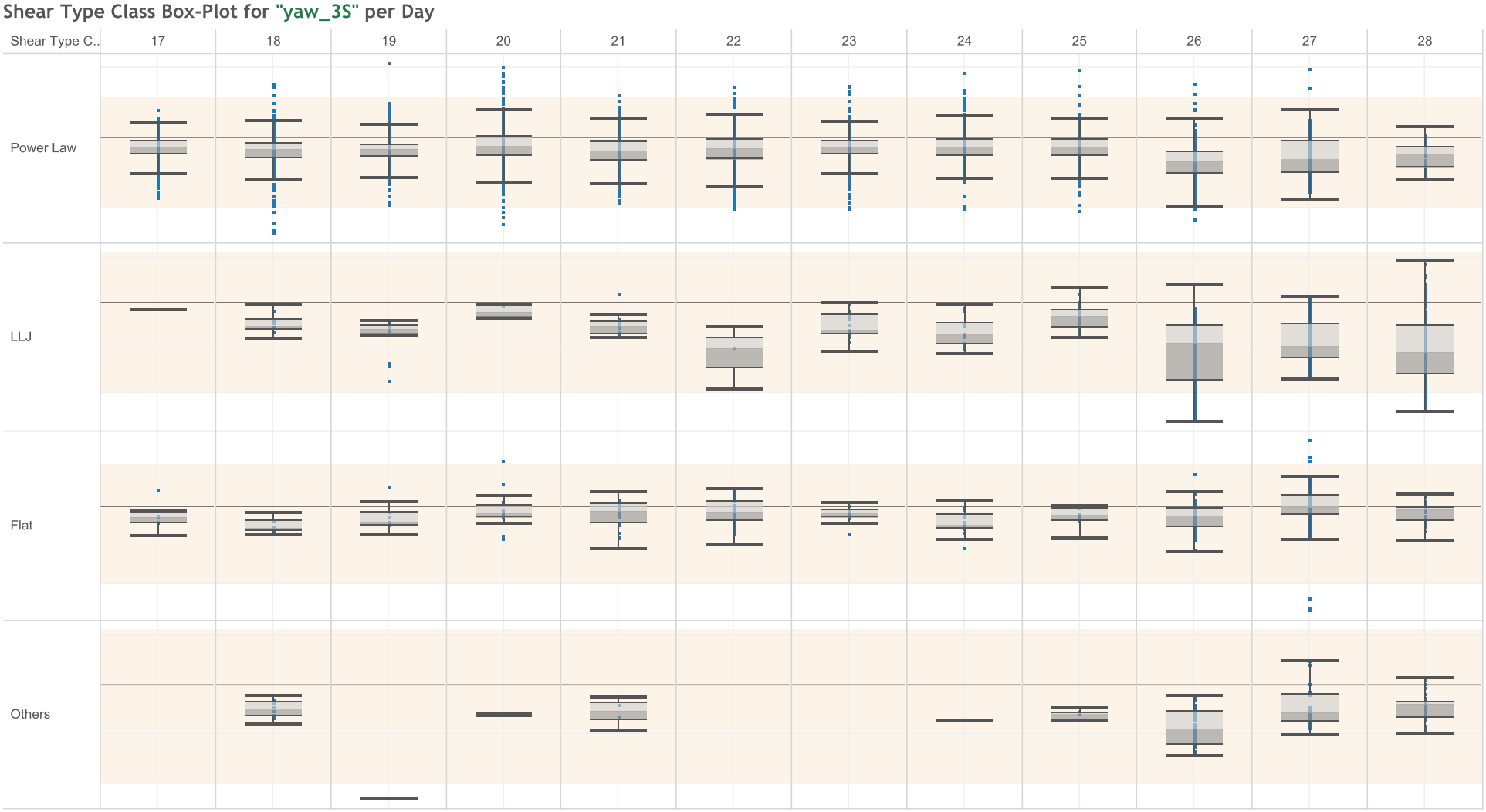


FIGURE 28 BOX PLOT FOR YAW_3S BY DAY.

What about the correlation. The following charts show a mutual correlation or connection between each variable.

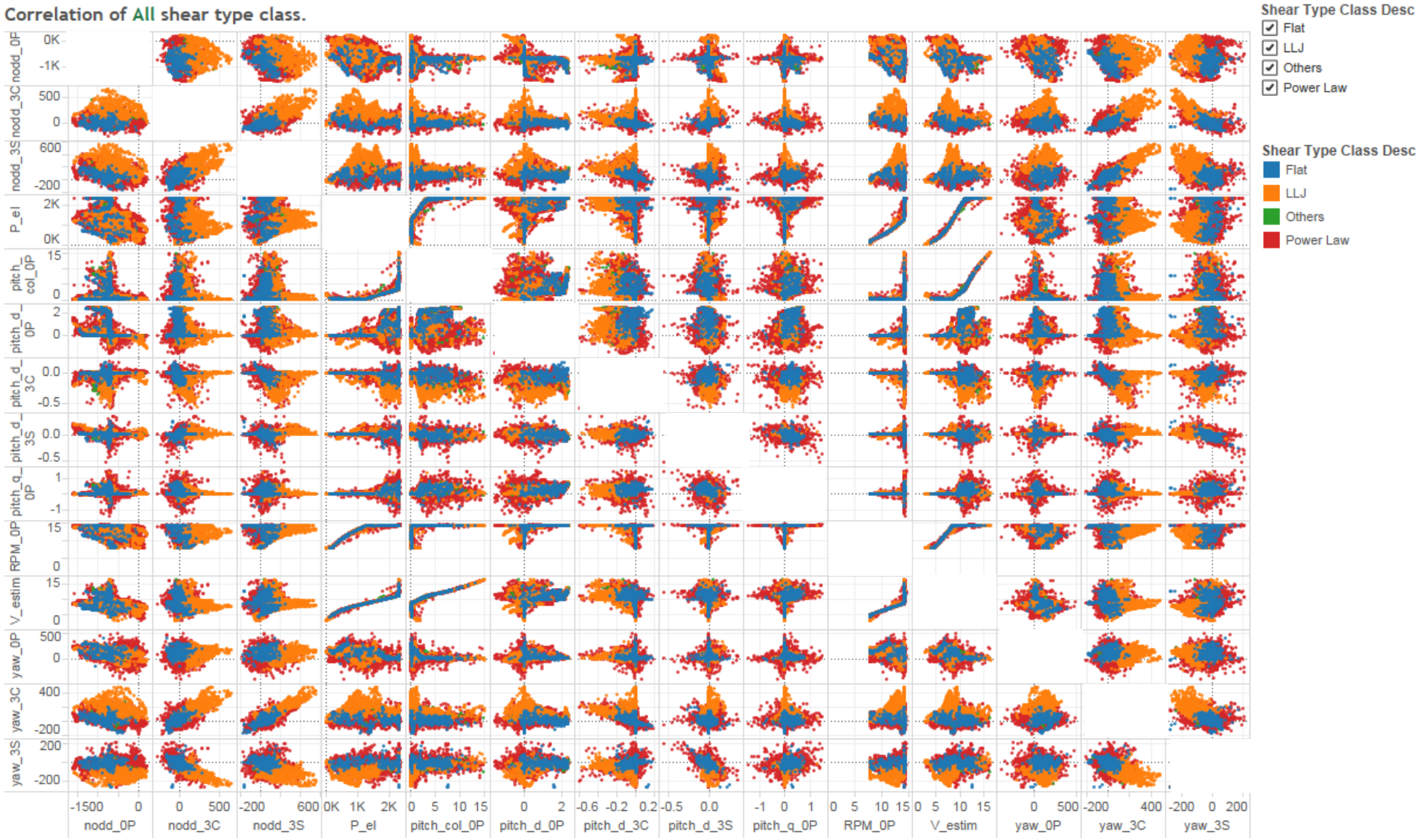


FIGURE 29 CORRELATION FOR ALL SHEAR TYPE CLASSES.

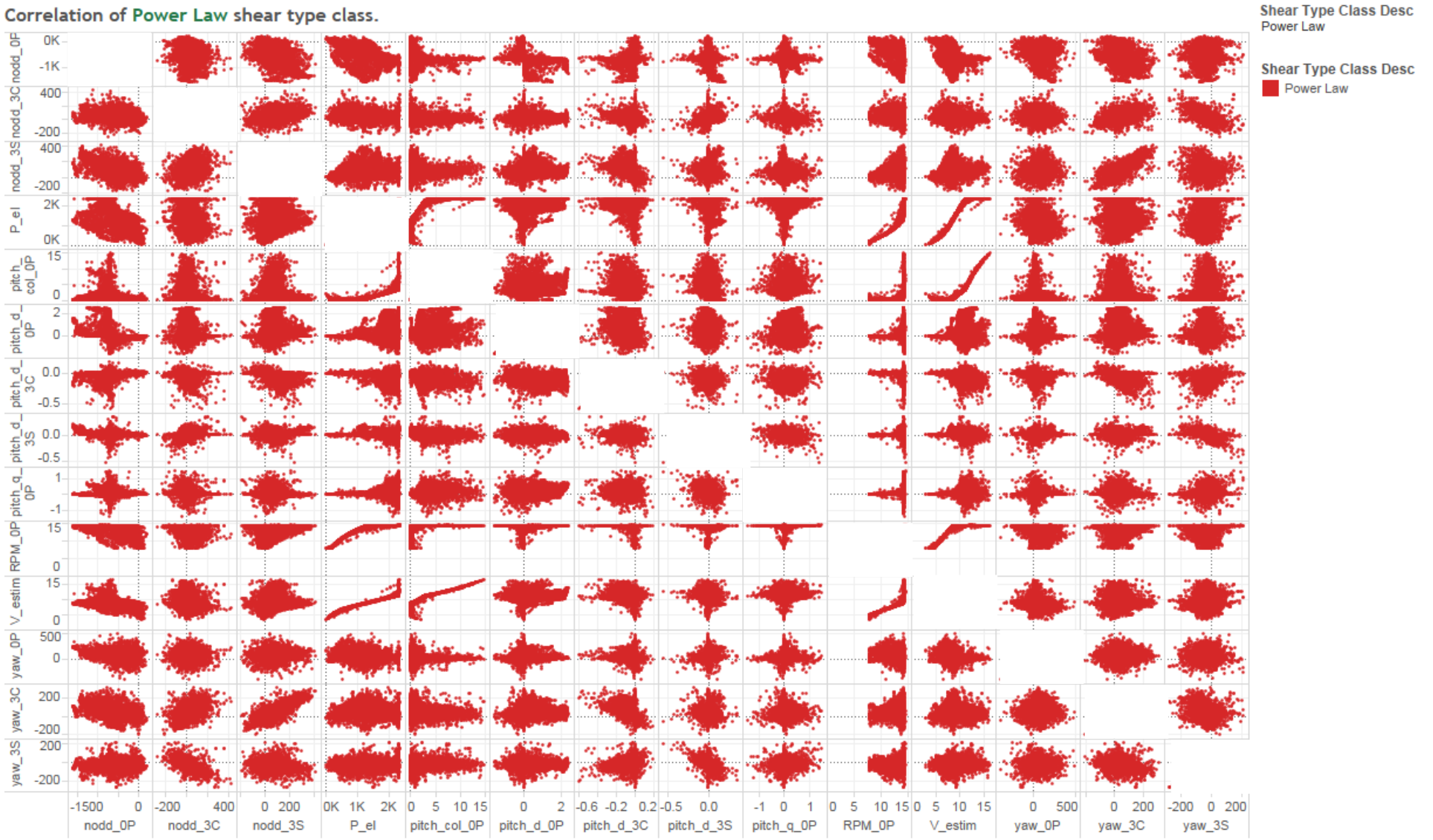


FIGURE 30 CORRELATION FOR POWER LAW SHEAR TYPE CLASS.

Correlation of LLJ shear type class.

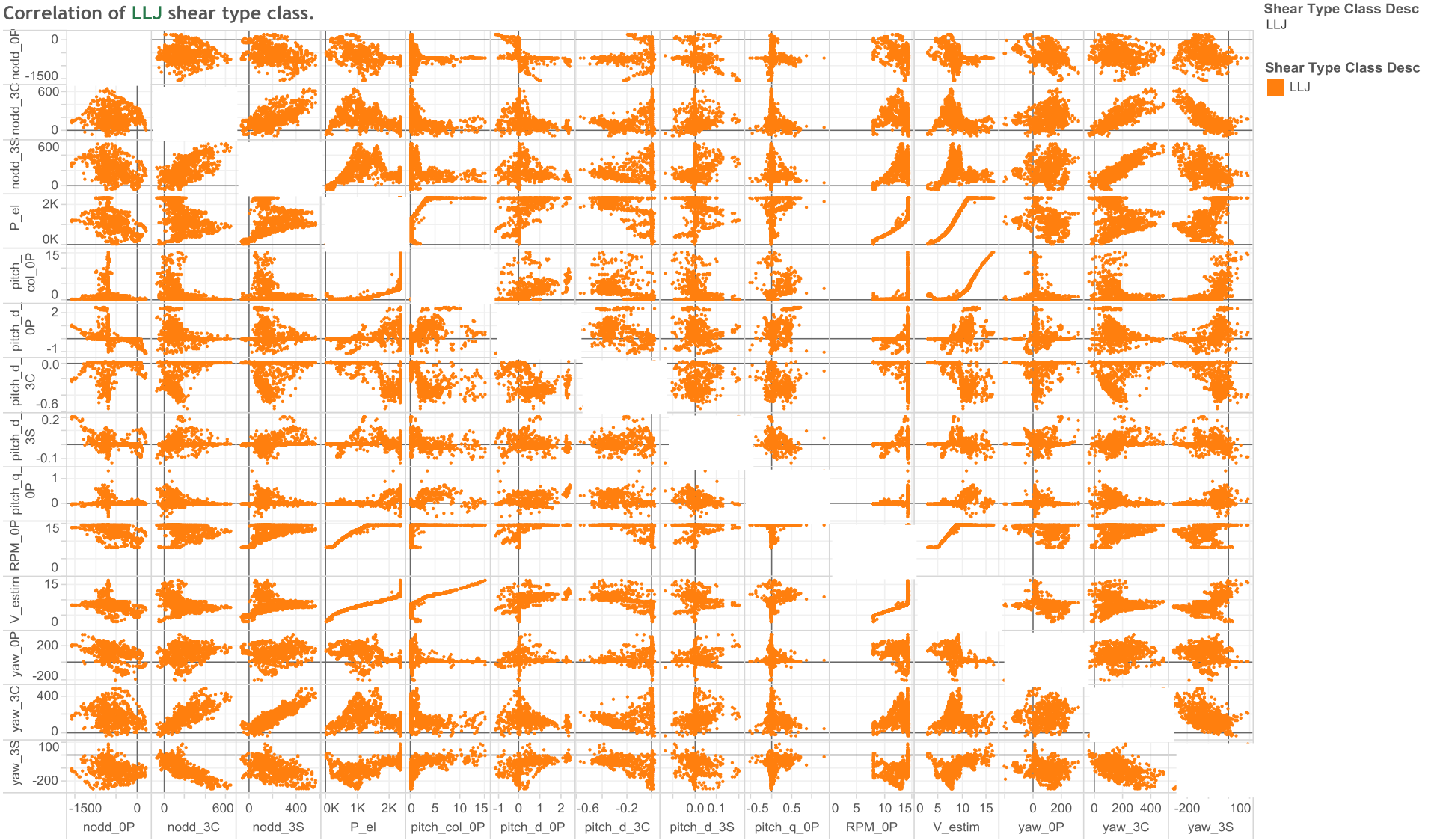


FIGURE 31 CORRELATION FOR LLJ SHEAR TYPE CLASS.

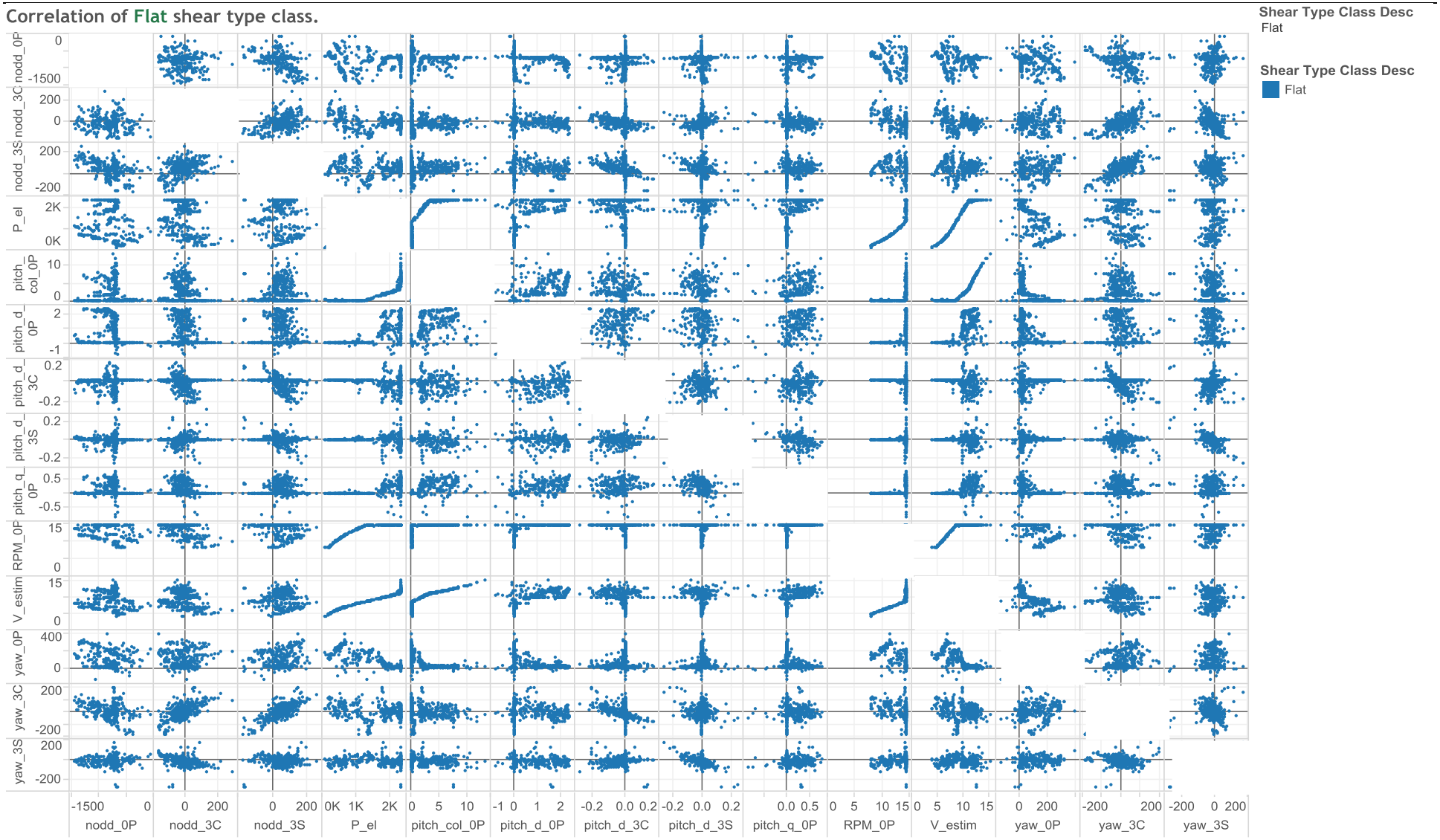


FIGURE 32 CORRELATION FOR FLAT SHEAR TYPE CLASS.

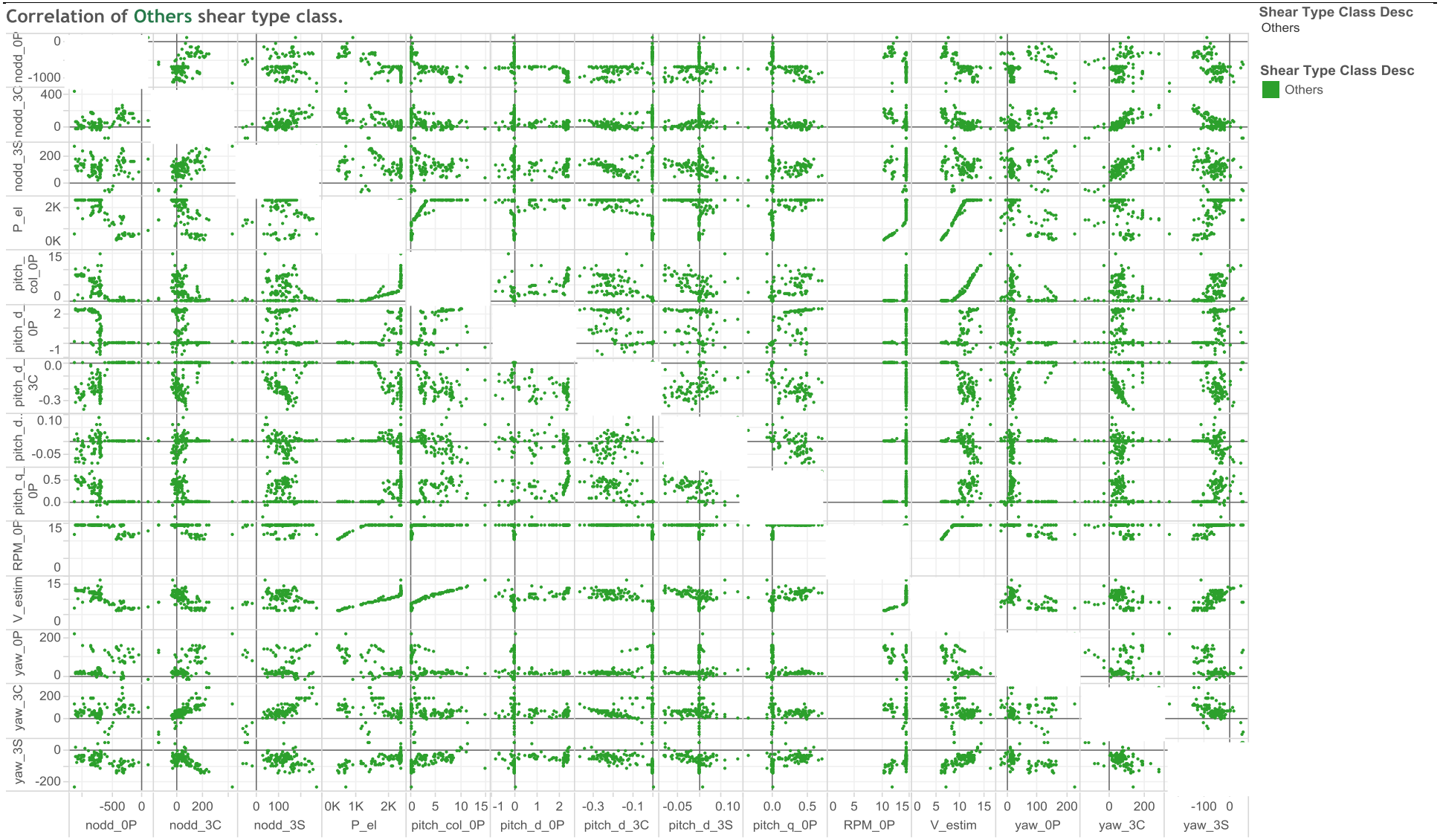


FIGURE 33 CORRELATION FOR OTHERS SHEAR TYPE CLASS.

How much are correlated the variables? Next chart show if there is a strong or weak correlation between variables. The color range varies from dark red (strong negative correlation), over white (no correlation), to dark blue (strong positive correlation).

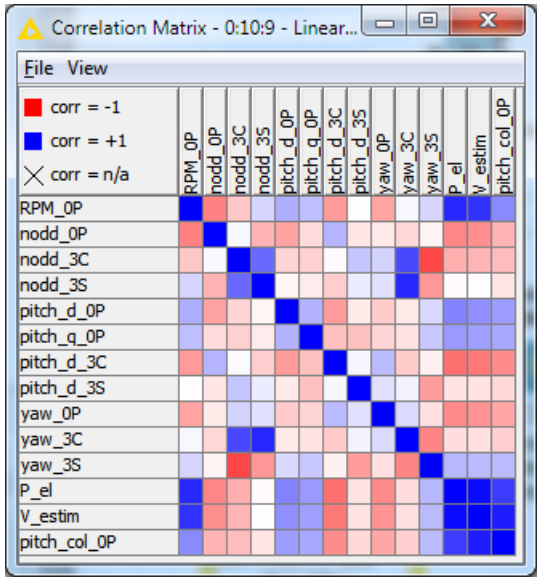


FIGURE 34 CORRELATION BETWEEN SHEAR TYPE CLASS.

	RPM_0P	nodd_0P	nodd_3C	nodd_3S	pitch_d_0P	pitch_q_0P	pitch_d_3C	pitch_d_3S	yaw_0P	yaw_3C	yaw_3S	P_el	V_estim	pitch_col_0P
RPM_0P	1	-0.489881	-0.217764	0.166532	0.317437	0.252289	-0.396089	-0.009761	-0.362436	0.032400	0.163534	0.847775	0.803124	0.457272
nodd_0P	-0.489881	1	0.028683	-0.293095	-0.364921	-0.142116	0.293677	-0.101411	-0.084940	-0.155436	-0.050424	-0.475605	-0.443863	-0.292362
nodd_3C	-0.217764	0.028683	1	0.587782	-0.163174	-0.183376	0.014212	0.227009	0.176791	0.716318	-0.727280	-0.318744	-0.292113	-0.262034
nodd_3S	0.166532	-0.293095	0.587782	1	-0.039636	-0.076832	-0.194697	0.076364	0.119754	0.851559	-0.407134	-0.014260	-0.006516	-0.107500
pitch_d_0P	0.317437	-0.364921	-0.163174	-0.039636	1	0.296721	-0.393338	-0.081131	-0.205865	-0.088192	0.150209	0.485963	0.439905	0.385029
pitch_q_0P	0.252289	-0.142116	-0.183376	-0.076832	0.296721	1	-0.247423	-0.248417	-0.166746	-0.107858	0.217531	0.400638	0.376301	0.341552
pitch_d_3C	-0.396089	0.293677	0.014212	-0.194697	-0.393338	-0.247423	1	0.035046	0.266358	-0.202313	-0.055226	-0.562321	-0.530671	-0.449337
pitch_d_3S	-0.009761	-0.101411	0.227009	0.076364	-0.081131	-0.248417	0.035046	1	0.118182	0.048598	-0.389300	-0.110214	-0.115631	-0.152969
yaw_0P	-0.362436	-0.084940	0.176791	0.119754	-0.205865	-0.166746	0.266358	0.118182	1	0.145939	-0.126254	-0.460476	-0.428659	-0.349410
yaw_3C	0.032400	-0.155436	0.716318	0.851559	-0.088192	-0.107858	-0.202313	0.048598	0.145939	1	-0.483855	-0.131819	-0.125064	-0.189116
yaw_3S	0.163534	-0.050424	-0.727280	-0.407134	0.150209	0.217531	-0.055226	-0.389300	-0.126254	-0.483855	1	0.281046	0.271209	0.271444
P_el	0.847775	-0.475605	-0.318744	-0.014260	0.485963	0.400638	-0.562321	-0.110214	-0.460476	-0.131819	0.281046	1	0.960729	0.766599
V_estim	0.803124	-0.443863	-0.292113	-0.006516	0.439905	0.376301	-0.530671	-0.115631	-0.428659	-0.125064	0.271209	0.960729	1	0.886668
pitch_col_0P	0.457272	-0.292362	-0.262034	-0.107500	0.385029	0.341552	-0.449337	-0.152969	-0.349410	-0.189116	0.271444	0.766599	0.886668	1

TABLE 9 CORRELATION BETWEEN SHEAR TYPE CLASS

Methods

So far, previous data exploration helped to identify the behavior or the data, how much was collected, what are its values, etc. Also, helped to define a strategy to follow when classification/predictive algorithms are applied (e.g remove or not outliers, remove or not 4 days of data, etc.).

Highlights:

- In the first scenario (1st goal), the output **Y (ShearTypeClass) is nominal**
- In the second scenario (2nd goal), the output **Y (*shear* (α)) is numeric**
- There are some confounders when data is analyzed:
    - The features m58, m103 and m122 are not used
    - About 79% of data is classified as "Power Law", which could dominate the analysis.
    - Few data for days 17 and 28, versus other days, that could lead us to think that those days something wrong happened, or they started to collect data at some point of day first and stop collecting data at some point of last day
    - Days 26, 27 y 28 could be affected by the outliers of “LLJ” shear type class.
    - Looks weird that last three days of data, most of they were classified "LLJ", "Flat" or "Others", and few of them as "Power Law"
    - The variance could be affected by the mix of shear types.
    - Variables have different scales. Normalize data or remove outliers is recommended.
    - There are some variables with strong correlation. Variables can be reduced and just select one variable from those which are correlated.
    - Some variables are having many values around ZERO. Subject matter experts need to be consulted.

The next sections describe what models are used, references to those highlights and obtained results.

Modeling

This section is structured based on the goals to accomplish (see above section “Business Understanding” for details), briefly the goals are:

1. Part A. Develop and evaluate a classifier to see how well a speed profile can be determined from the load sensor data.
2. Part B. Create a predictive model to estimate the actual *shear* (α).
3. Part C. For speed profiles labeled as 1, 2, and 3, how might you create a model to estimate these profiles?

Note: For the goal 3 (Part C), only an explanation is included. The solution can be easily implemented using the objects used for goals 1 and 2. See details in Part C section below.

To see the details of the programing, open/import the file named "***Wind Shear Analysis Full.knwf***" into the KNIME app environment. This file is included into the deliverables. Next picture shows the process flow followed to analyze the data, the tests applied to several learning models and the selection of the model giving the best performance.

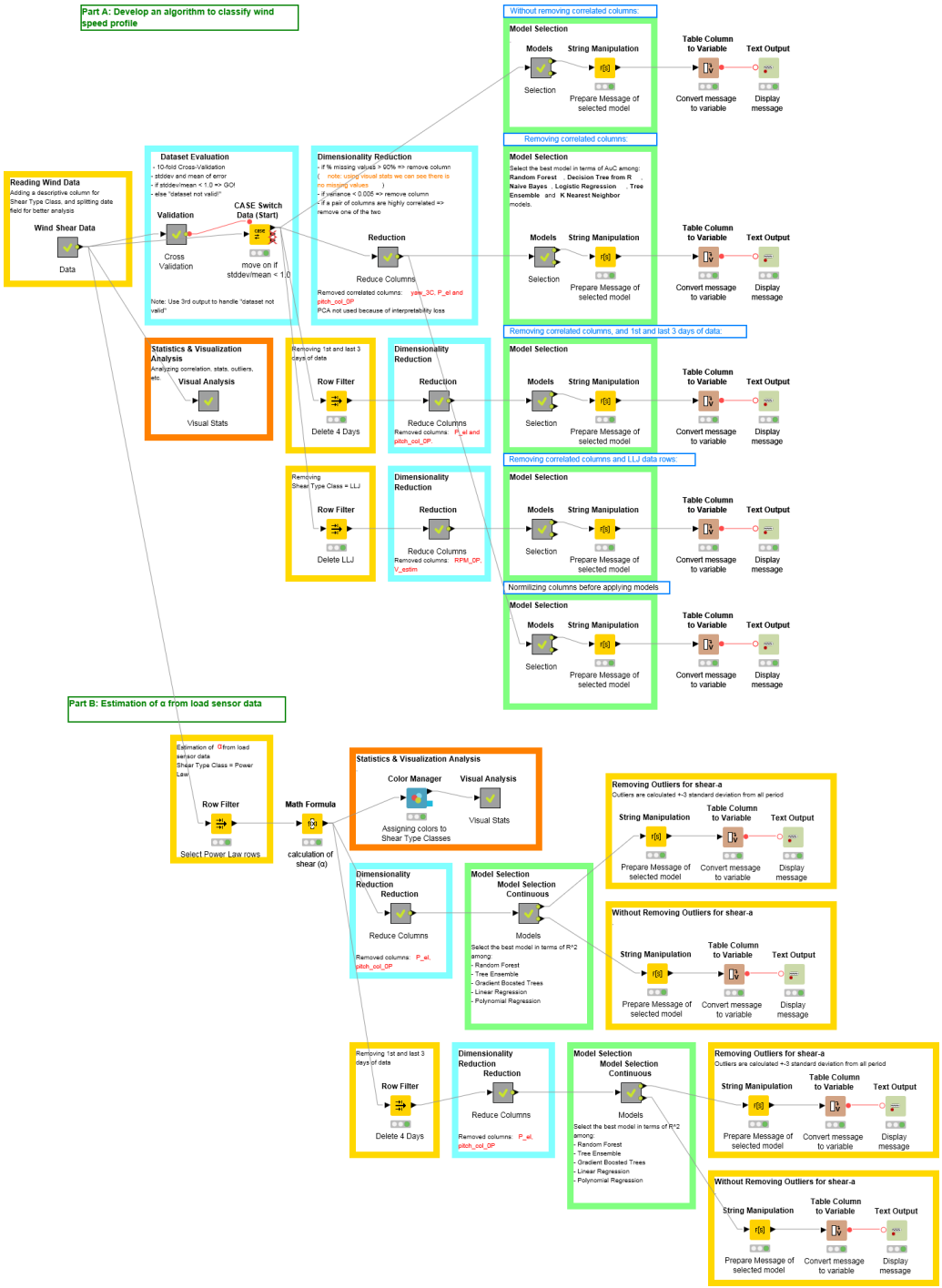


FIGURE 35 KNIME PROCESS FLOW FOR WIND SHEAR CASE STUDY ANALYSIS

On the previous picture, there is two sections: **Part A**, deal with the goal to develop and evaluate a classifier to see how well a ***speed profile*** can be determined from the load sensor data (trying to predict results in a discrete output). **Part B**, programming to select a predictive model to estimate the actual *shear* (α) (trying to predict results within a continuous output).

Note: It is important to differentiate if the output of model is continuous or discrete in order to identify which models to apply.

KNIME Process Flow Explanation – Part A

Data Preparation.

This node prepares the data. The output of this node is used for Part A and Part B.

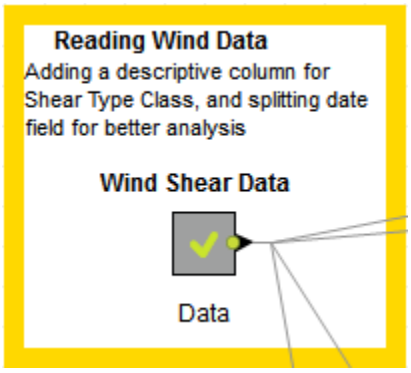
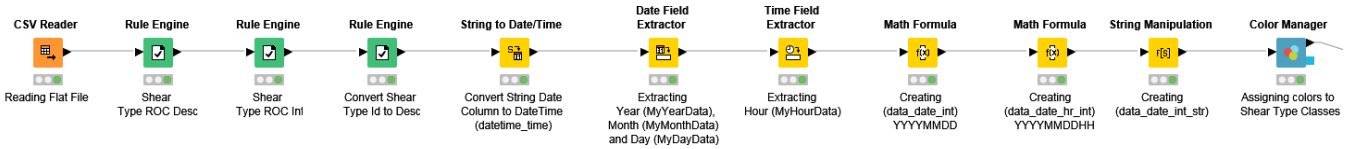
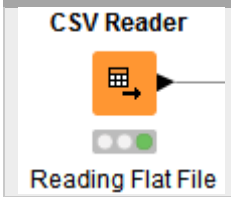
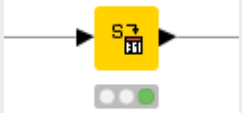
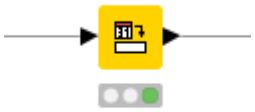
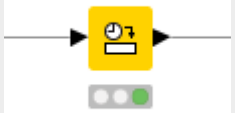


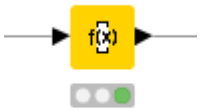
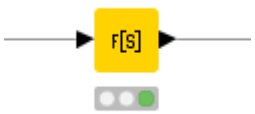
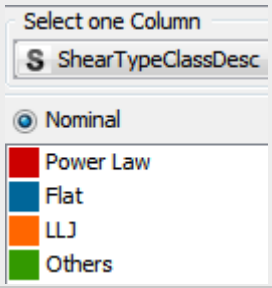
FIGURE 36 READING RAW DATA AND PREPARATION

It reads the flat file, adds new columns (useful to analyze data), converts data types and assign color to each "Shear Type Class".



Node	Description
	It reads the flat file "WindFarm_2min_AnalyticsEngineer_wWS.csv". **Note**: when you import or open the file "Wind Shear Analysis Full.knwf", could be requested to specify where is located your data file, so you need to update this node and include the path and data filename.

Rule Engine  Shear Type ROC Desc	It creates a string column “ShearTypeClassROCDesc”. This assign their value as: If ShearTypeClass = 0 then "Power Law" If ShearTypeClass <> 0 then "No Power Law" It is useful to evaluate the models in terms of AuC.
Rule Engine  Shear Type ROC Int	It creates an integer column “ShearTypeClassROCInt”. This assign their value as: If ShearTypeClass = 0 then 1 If ShearTypeClass <> 0 then 0 It is useful to evaluate the models in terms of AuC.
Rule Engine  Convert Shear Type Id to Desc	It creates a string column “ ShearTypeClassDesc”. This assign their value as: If ShearTypeClass = 0 then "Power Law" If ShearTypeClass = 1 then "LLJ" If ShearTypeClass = 2 then "Flat" If ShearTypeClass = 4 then "Others"
String to Date/Time  Convert String Date Column to DateTime (datetime_time)	It creates a Date/Time column “datetime_time”. Converts a string (datetime column) to Date/Time.
Date Field Extractor  Extracting Year (MyYearData), Month (MyMonthData) and Day (MyDayData)	It creates three columns: MyYearData = year of datetime_time. MyMonthData = month of datetime_time. MyDayData = day of datetime_time.
Time Field Extractor  Extracting Hour (MyHourData)	It creates the integer column MyHourData with the hour of datetime_time.

Math Formula  Creating (data_date_int) YYYYMMDD	It creates the numeric column data_date_int with datetime_time in format YYYYMMDD
Math Formula  Creating (data_date_hr_int) YYYYMMDDHH	It creates a numeric column data_date_hr_int with datetime_time in format YYYYMMDDHH.
String Manipulation  Creating (data_date_int_str)	It creates the string column data_date_int_str. It converts the numeric data_date_int field to string.
Color Manager  Assigning colors to Shear Type Classes	It assigns colors to Shear Type Class. Useful to visualization analysis. It is assigned as: 

Here is how it looks the original data table and additional control columns. Note its type and bounds.

Column	Column Type	Column Index	Lower Bound	Upper Bound
datetime	String	0	?	?
m38	Number (double)	1	0.48029	15.978
m58	Number (double)	2	1.4018	17.478
m78	Number (double)	3	2.653	18.867
m103	Number (double)	4	2.7397	18.845
m122	Number (double)	5	2.8878	18.637
RPM_0P	Number (double)	6	7.9608	14.618
nodd_0P	Number (double)	7	-1593.1	205.97
nodd_3C	Number (double)	8	-267.87	652.53
nodd_3S	Number (double)	9	-183.16	561.63

pitch_d_0P	Number (double)	10	-1.5862	2.433
pitch_q_0P	Number (double)	11	-1.44	1.493
pitch_d_3C	Number (double)	12	-0.59653	0.1876
pitch_d_3S	Number (double)	13	-0.54678	0.35374
yaw_0P	Number (double)	14	-387.52	489.13
yaw_3C	Number (double)	15	-201.44	487.16
yaw_3S	Number (double)	16	-274.14	208.73
P_el	Number (double)	17	3.384	2359.4
V_estim	Number (double)	18	2.7557	16.457
pitch_col_0P	Number (double)	19	0.040203	15.102
ShearTypeClass	Number (integer)	20	0	3
ShearTypeClassROCDesc	String	21	?	?
ShearTypeClassROCInt	Number (integer)	22	0	1
ShearTypeClassDesc	String	23	?	?
datetime_time	Date and Time	24	2015-06-17T10:30:00	2015-06-28T14:39:00
MyYearData	Number (integer)	25	2015	2015
MyMonthData	Number (integer)	26	6	6
MyDayData	Number (integer)	27	17	28
MyHourData	Number (integer)	28	0	23
data_date_int	Number (double)	29	2.02E+07	2.02E+07
data_date_hr_int	Number (double)	30	2.02E+09	2.02E+09
data_date_int_str	String	31	?	?

TABLE 10 SOURCE TABLE WITH ADDITIONAL COLUMNS

Dataset Evaluation

This node is used to evaluate if the dataset is valid or not. It uses ***Cross-Validation*** model. Based on coefficient of variation (standard deviation / mean) from the data source, it evaluates if the coefficient is less than 1.0. In case data is having a great variation (coefficient > 1) is preferred to review again the system used to collect the data, and not continue with the analysis. This could be a redundant step if we already evaluate the system used to collect the data. In addition, this evaluation helps to identify if data could produce a lot of "noise".

The process flow continues if data is valid (first output port of CASE Switch component – see the CASE Switch node in the next picture). Notice that two output ports have an X red, which indicates that only first condition was true (valid data), invalid data output is in red. If you want to control when the data is not valid, you can link these outputs to handle the error appropriately (for example, send an email error). Invalid condition is not handled into the KNIME process programmed.

So, the first output port from the CASE switch is the original table, if it passes the validation data steps.

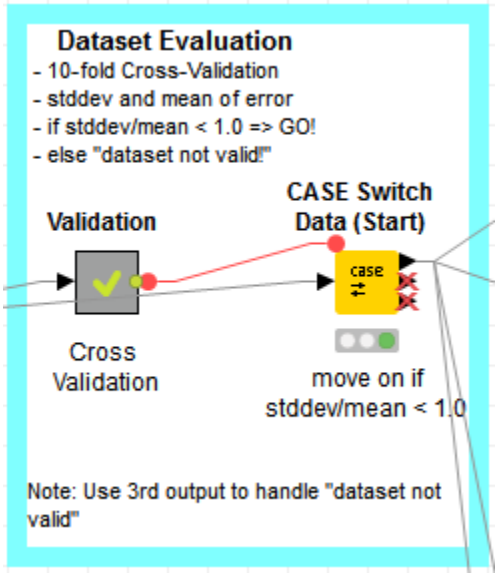


FIGURE 37 DATASET EVALUATION

VALIDATION NODE

The detail of the dataset evaluation node is managed in the process flow drawn next picture.

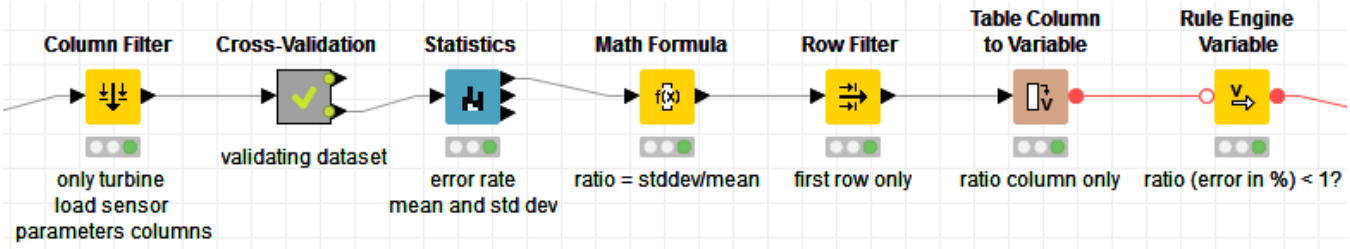


FIGURE 38 VALIDATION NODE PROCESS

Steps:

1. Column Filter. Only X variables and Shear Type Class columns are selected.
2. Cross-Validation. It calculated the Error rates for all iterations, using Decision Tree model for each iteration.
3. Statistics. This node calculates statistical moments such as standard deviation and mean between other calculations. It calculates statistics from the output of Cross-Validation node, which are the Error rates for all iterations applied into Cross-Validation node. See the Cross-Validation Node steps below.
4. Math Formula. It calculates the ratio (coefficient of variation = standard deviation / mean). The output of this node has 3 rows (Error in %, Size of the Test Set and Error Count).
5. Row Filter. It takes only the first row from the output of Math Formula node, which is the Error in %.
6. Table Column to Variable. It takes the column 'ratio' from the output of Row Filter.
7. Rule Engine Variable. It creates a variable (prediction) with the evaluation of the ratio. It returns 0 if ratio error is less than 0, otherwise 2.

CROSS-VALIDATION NODE

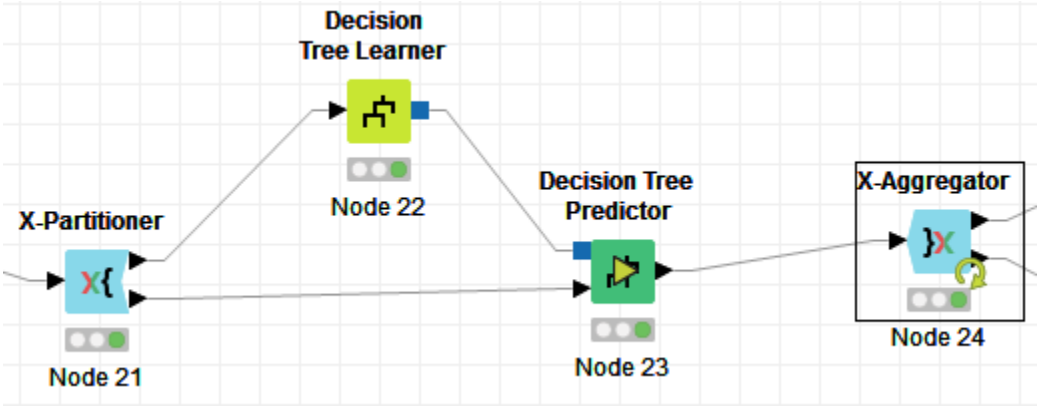


FIGURE 39 CROSS VALIDATION NODE PROCESS

Steps:

1. X-Partitioner. It creates a loop with 10 iterations.
2. Decision Tree Learner. It is executed into each iteration. It executes the Decision Tree model on the dataset.
3. Decision Tree Predictor. It is also executed into each iteration after Decision Tree Learner node. It predicts the class value for new patterns. It adds a new column with the prediction.
4. X-Aggregator. It collects the result from a predictor node, compares predicted class and real class and outputs the predictions for all rows and the iteration statistics. The second output is what we are interested, it gives the Error rates for all iterations.

Dimensionality Reduction Node.

A good practice is to discard several variables if they are highly (linearly) correlated to other variables before doing PCA (Principal Component Analysis). Nevertheless, also is evaluated the model selection without reducing these variables in order to see if they affect in the model selection.

The columns can be reduced based on:

- too many missing values (as shown above, there is no variables with missing values), so no validation was automated into KNIME to handle it.
- too low variance, and
- too high correlation with another column(s)

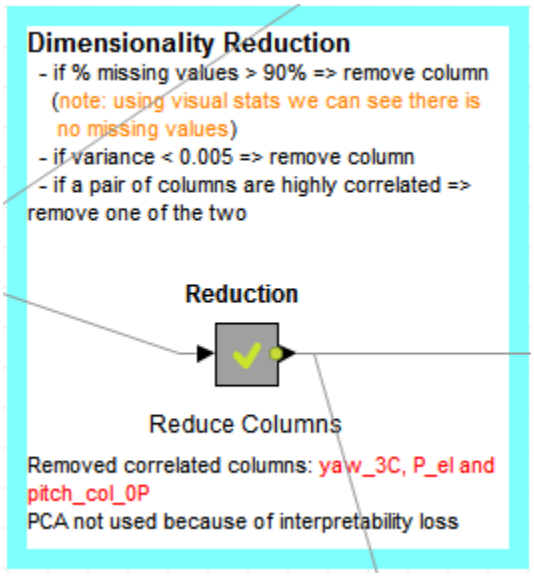


FIGURE 40 DIMENSIONALITY REDUCTION NODE

Inside the Reduction Node is applied the Linear Correlation to the variables.

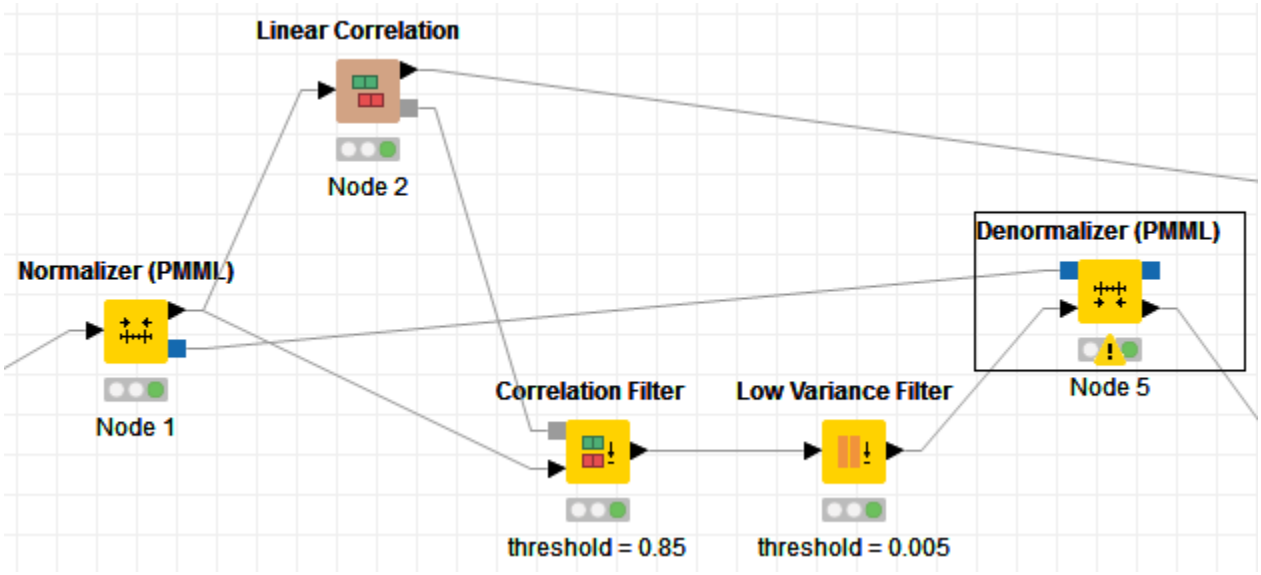


FIGURE 41 CORRELATION AND VARIANCE NODE

Steps:

1. Normalizer. This node normalizes the values of all (numeric) columns. The Min-max normalization is used.
2. Linear Correlation. Calculates for each pair of selected columns a correlation coefficient, i.e. a measure of the correlation of the two variables.
3. Correlation Filter. This node uses the model as generated by a Correlation node to determine which columns are redundant (i.e. correlated) and filters them out. The output table will contain the reduced

set of columns (threshold = 0.85; i.e. if a pair of columns are highly correlated then remove one of the two).

4. Low Variance Filter. Filters out double-compatible columns, whose variance is below a user defined threshold (threshold = 0.005; i.e. if variance < 0.005 then remove column).
5. Denormalizer. This node denormalizes the input data according to the normalization parameters as given in the Normalizer node.

Statistics and Visualization Analysis node.

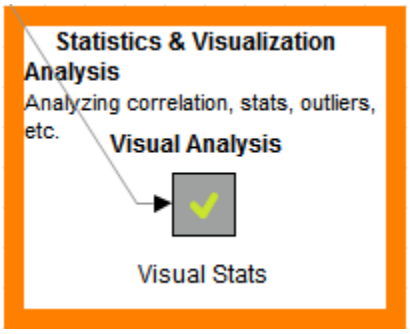


FIGURE 42 STATISTICS AND VISUALIZATION ANALYSIS NODE

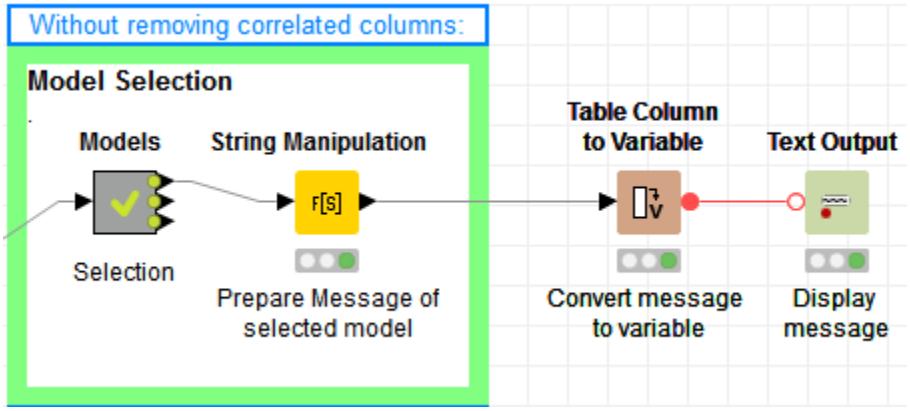
This node helps to analyze statistical information such as minimum, maximum, mean, standard deviation, variance, median, overall sum, number of missing values and row count across all numeric columns, and counts all nominal values together with their occurrences. In addition, it includes One Way ANOVA node, Box-Plot node, Scatter-Plot node and linear correlation node.

This information was already explained above this document. See Data Understanding and Data Preparation sections. For further details of this node, open it in KNIME.

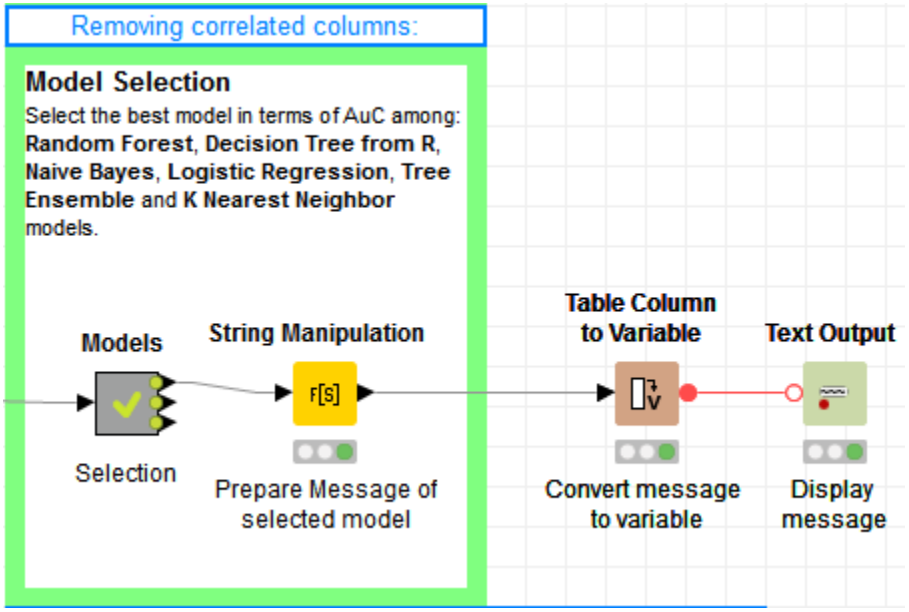
Model Selection node for discrete output

The **Model Selection** node is the core for the classification or prediction model selection. In order to identify the best model, they were tested under different scenarios, which are:

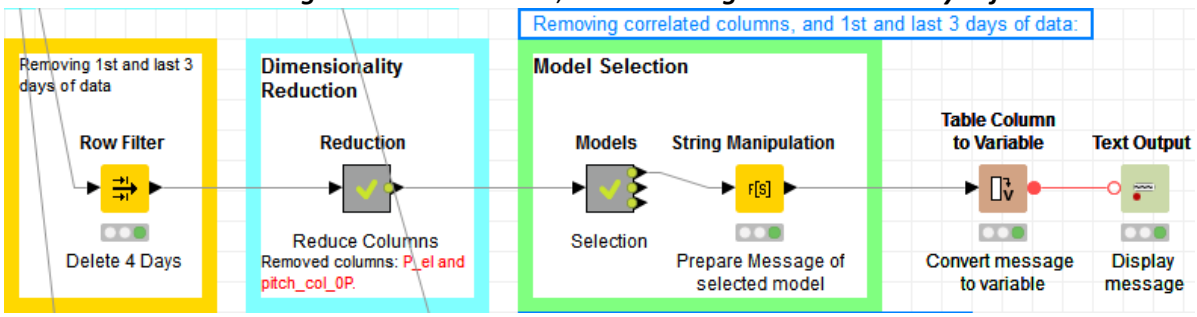
A. Test the models ***without removing correlated columns***.



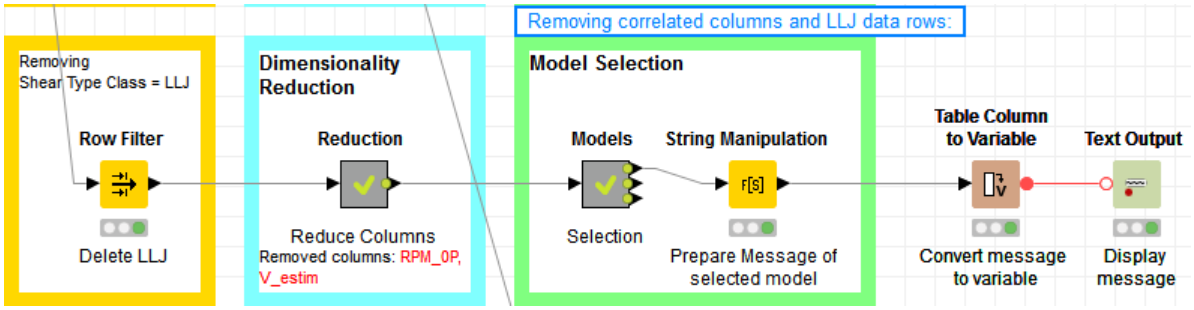
B. Test the models ***removing correlated columns***.



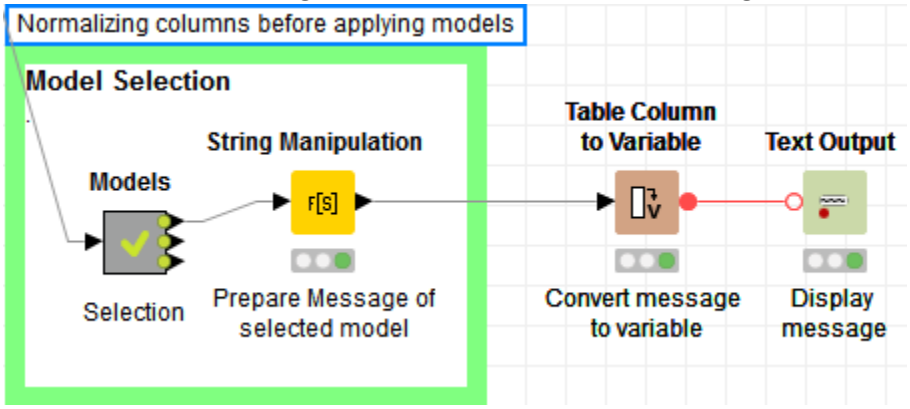
C. Test the models ***removing correlated columns***, and ***removing 1st and last 3 days of data***.



D. Test the models ***removing correlated columns*** and ***LLJ data rows***.



E. Test the models ***removing correlated columns*** and ***normalizing columns*** before applying models.



Note: the nodes "***String Manipulation***", "***Table Column to Variable***" and "***Text Output***" are used to display the message containing the model selected and the its value. Here is an example of this message:

Label
Selected Model: Tree Ensemble Power Law with AuC = 0.9308844475834358.

Model Selection node in details for discrete output

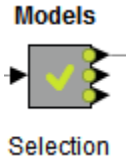


FIGURE 43 MODEL SELECTION NODE FOR DISCRETE

This node selects the best model in terms of AuC and in terms of Accuracy among following models (why they? – see appendix):

- Random Forest
- Decision Tree (using R)
- Naive Bayes
- Logistic Regression
- Tree Ensemble
- K Nearest Neighbor models.

Notice this node has one input and tree outputs:

- Input: It takes the dataset to be analyzed according scenarios described above.
- Output 1: The best model in terms of **AuC**.
  Note: In order to select the best model, it is used the column ShearTypeClassROCInt, which takes 1 for "Power Law" class (for which we are calculating the AuC) and 0 for the rest of the other classes.
- Output 2: It gives the original table with additional columns that contain the predicted value depending on the model tested.
- Output 3: It gives the model selected based on the **Accuracy**.

If we open the Models node, we can see the next nodes inside.

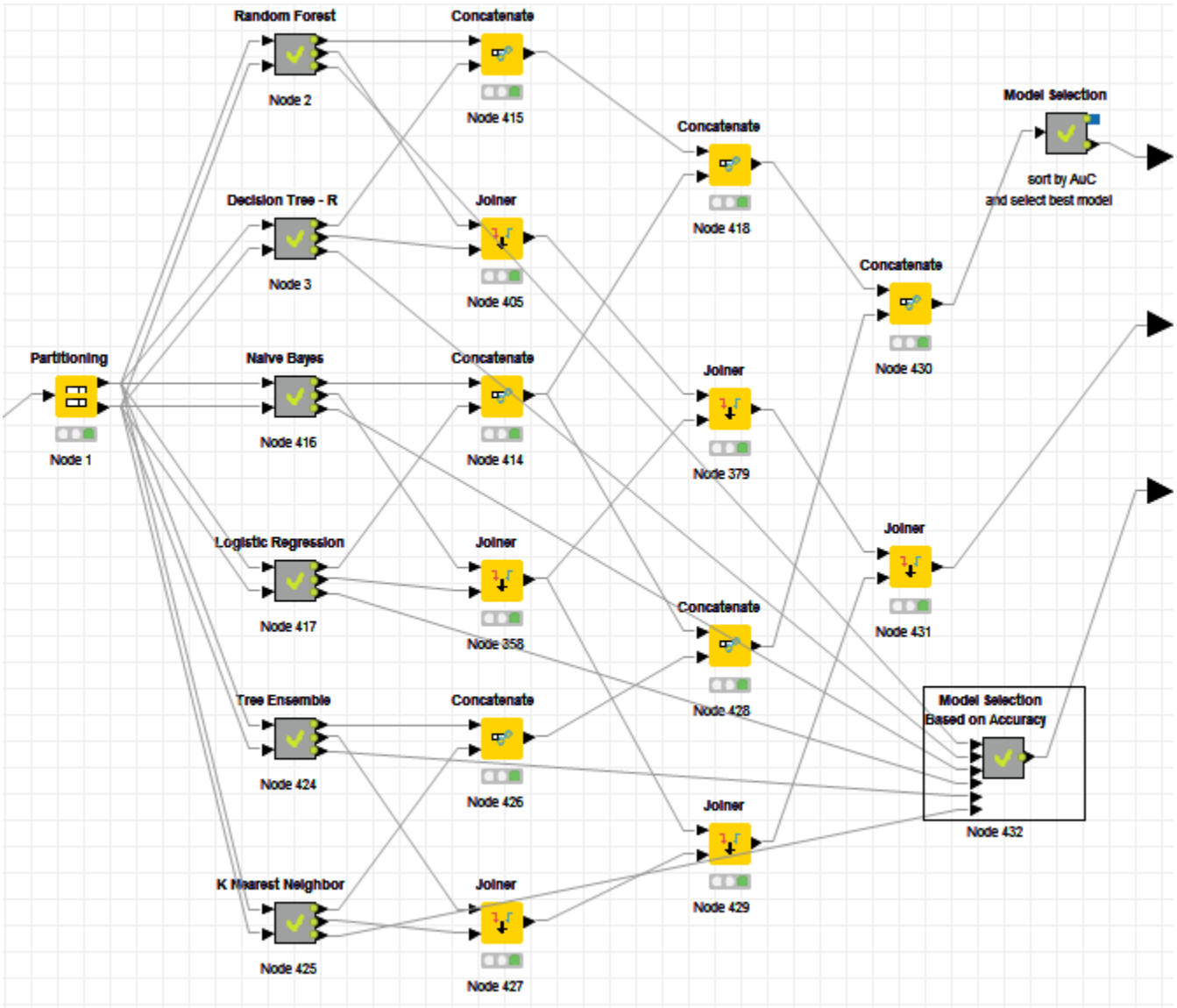
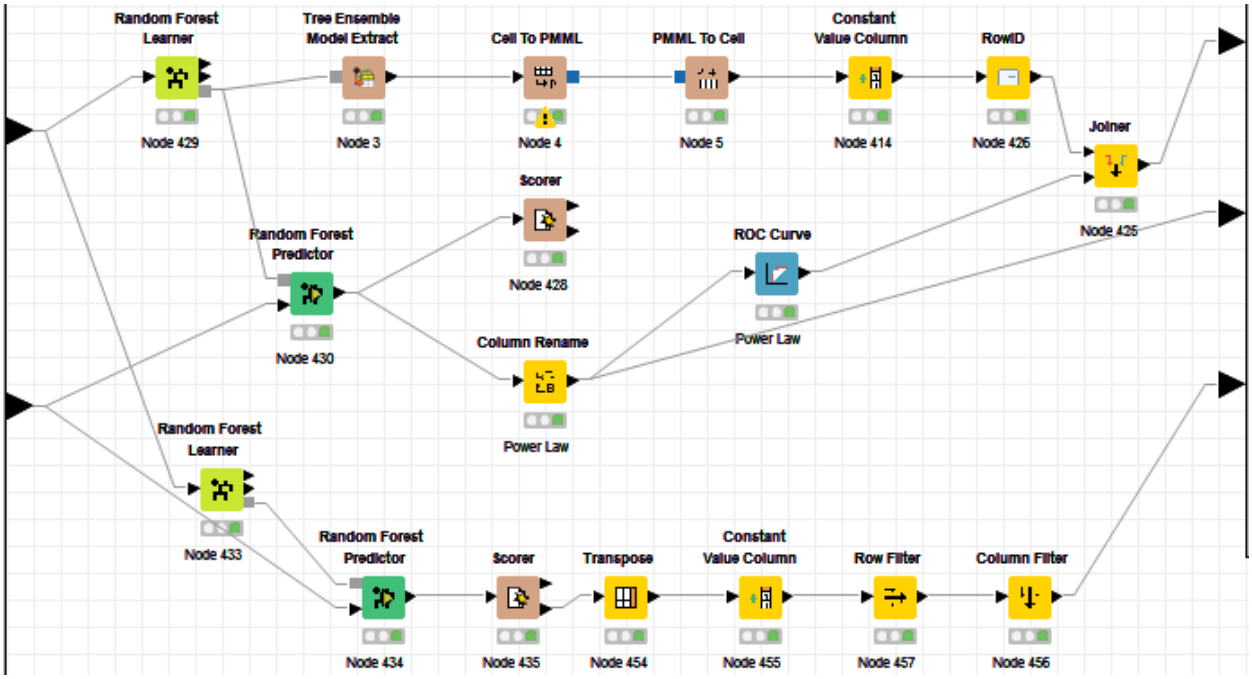


FIGURE 44 MODEL SELECTION NODE IN DETAILS

Model selection node steps and the models evaluated models:

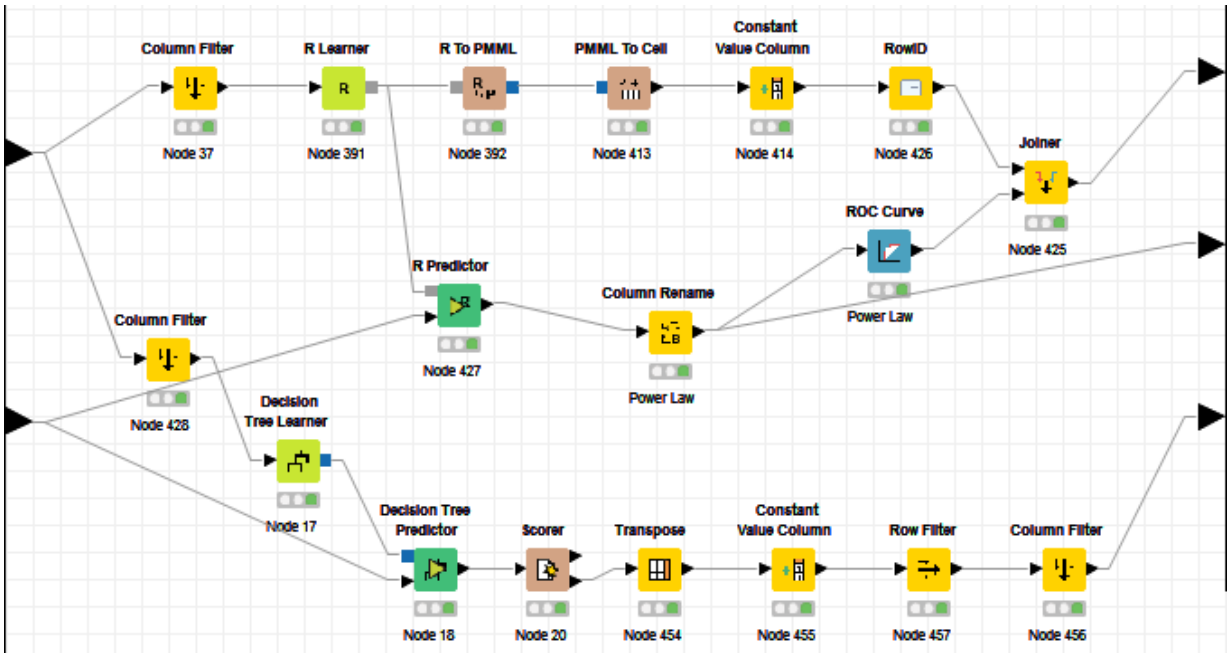
1. Partitioning: The input table is split into two partitions (i.e. row-wise), e.g. train and test data. The two partitions are available at the two output ports.
2. **Random Forest** contained nodes:



   a. Tree Ensemble Model Extract: Extracts individual decision trees from a tree ensemble model.
   b. Cell To PMML: Converts the PMML cell in the first Row to the PMML Port.
   c. PMML To Cell: Converts the PMML Port to a table containing the PMML cell.
   d. Constant Value Column: Adds a column containing a constant cell in each row.
   e. Joiner: Joins two tables
   f. RowID: Node to replace the RowID and/or to create a column with the values of the current RowID.
   g. Scorer: Compares two columns by their attribute value pairs.
   h. **Random Forest Learner**: Learns a random forest for classification. It uses ***ShearTypeClassROCDesc*** as target column.
   i. **Random Forest Predictor**: Predicts patterns according to a majority vote in a random forest model. It creates the predictive and probability columns for each class.
   j. Column Rename: Enables you to rename column names or to change their types.
   k. ROC Curve: Shows ROC curves
   l. **Random Forest Learner**: Learns a random forest for classification. It uses ***ShearTypeClassDesc*** as target column.
   m. **Random Forest Predictor**: Predicts patterns according to a majority vote in a random forest model. It creates the predictive and probability columns for each class.
   n. Scorer: Compares two columns by their attribute value pairs.
   o. Transpose: Transposes a table by swapping rows and columns.

p. Constant Value Column: Adds a column containing a constant cell in each row.
q. Column Filter: The Column Filter allows columns to be excluded from the input table.
r. Row Filter: Allows filtering of data rows by certain criteria, such as row ID, attribute value, and row number range.

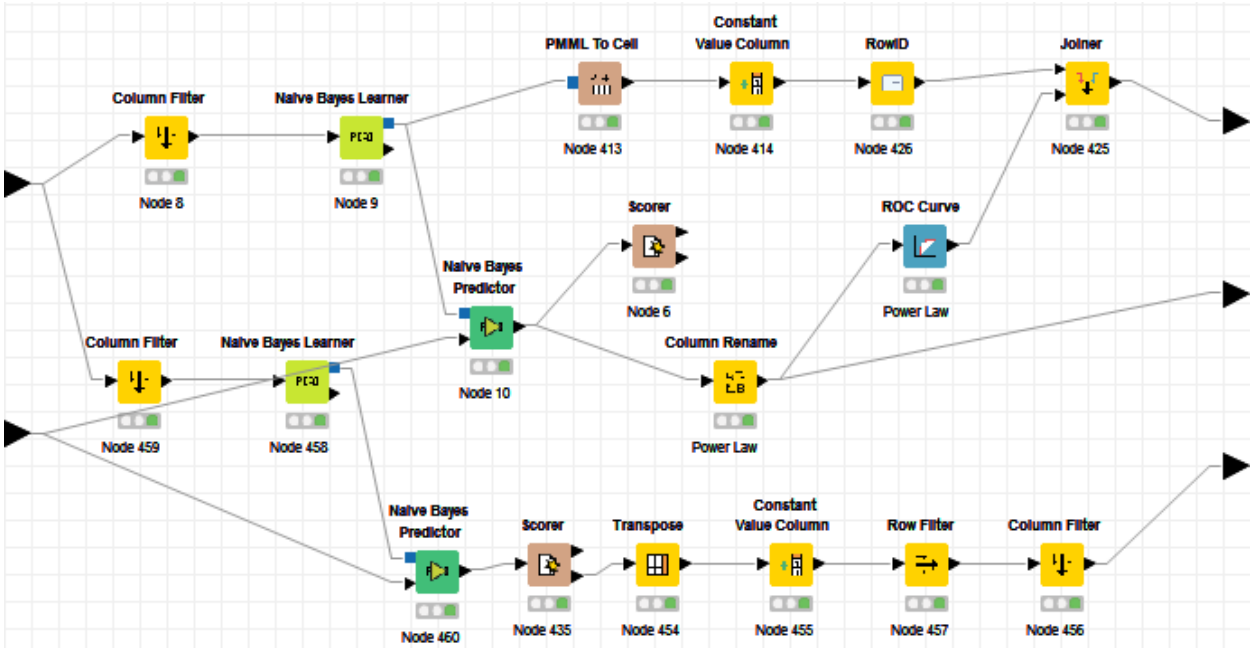
3. **Decision Tree - R** contained nodes:



a. **Decision Tree Learner**: Decision tree induction performed in memory. It uses ***ShearTypeClassDesc*** as target column.
b. **Decision Tree Predictor**: Uses an existing decision tree to compute class labels for input vectors. It creates the predictive and probability columns for each class.
c. Scorer: Compares two columns by their attribute value pairs.
d. Column Filter: The Column Filter allows columns to be excluded from the input table.
e. Column Rename: Enables you to rename column names or to change their types.
f. **R Learner**: Allows execution of R commands in a local R installation for building an R model. It uses ***ShearTypeClassROCDesc*** as target column.
g. R To PMML: Converts a given R object into a corresponding PMML object.
h. PMML To Cell: Converts the PMML Port to a table containing the PMML cell.
i. Constant Value Column: Adds a column containing a constant cell in each row.
j. ROC Curve: Shows ROC curves
k. Joiner: Joins two tables
l. RowID: Node to replace the RowID and/or to create a column with the values of the current RowID.
m. **R Predictor**: Allows execution of R code in a local R installation for predicting data attributes using new data and an existing model. It creates the predictive and probability columns for each class.
n. Column Filter: The Column Filter allows columns to be excluded from the input table.
o. Transpose: Transposes a table by swapping rows and columns.

p. Constant Value Column: Adds a column containing a constant cell in each row.
q. Column Filter: The Column Filter allows columns to be excluded from the input table.
r. Row Filter: Allows filtering of data rows by certain criteria, such as row ID, attribute value, and row number range.

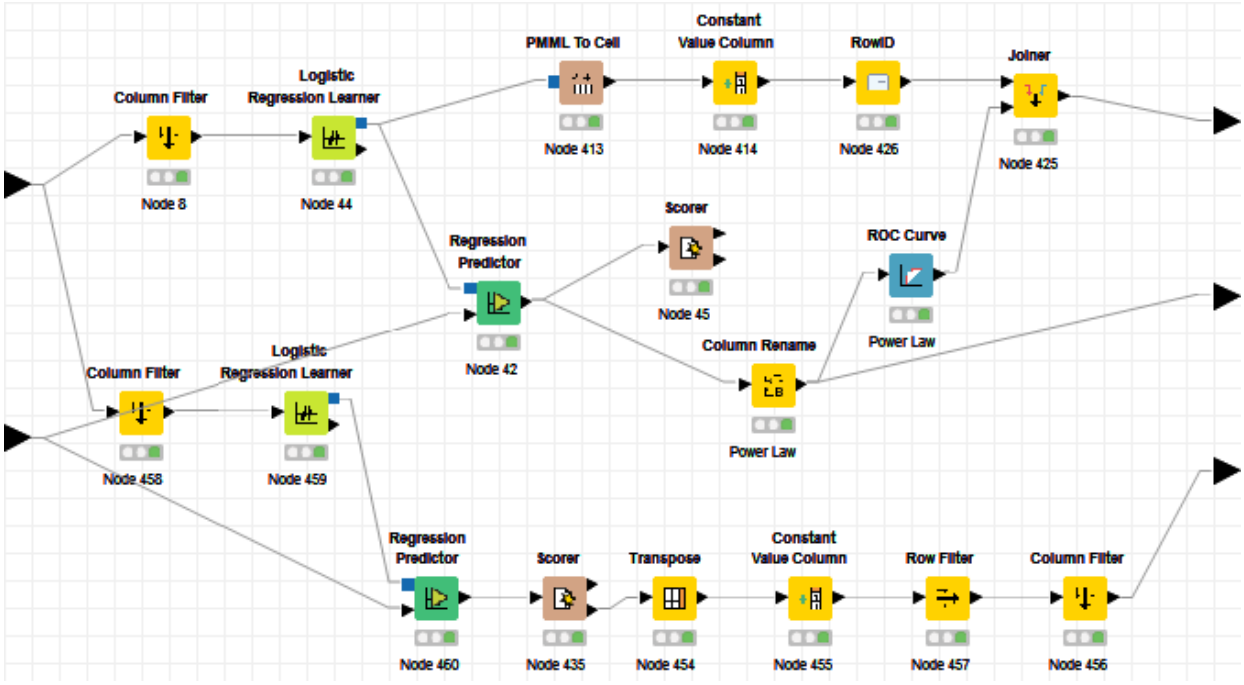
4. Joiner: Joins two tables
5. Concatenate: Concatenates two tables row-wise.
6. **Naive Bayes** contained nodes:



a. Scorer: Compares two columns by their attribute value pairs.
b. Column Filter: The Column Filter allows columns to be excluded from the input table.
c. **Naive Bayes Learner**: Creates a naive Bayes model from the given classified data. It uses ***ShearTypeClassROCDesc*** as target column.
d. **Naive Bayes Predictor**: Uses the PMML naive Bayes model from the naive Bayes learner to predict the class membership of each row in the input data. It creates the predictive and probability columns for each class.
e. Column Rename: Enables you to rename column names or to change their types.
f. PMML To Cell: Converts the PMML Port to a table containing the PMML cell.
g. Constant Value Column: Adds a column containing a constant cell in each row.
h. ROC Curve: Shows ROC curves
i. Joiner: Joins two tables
j. RowID: Node to replace the RowID and/or to create a column with the values of the current RowID.
k. Scorer: Compares two columns by their attribute value pairs.
l. Transpose: Transposes a table by swapping rows and columns.
m. Constant Value Column: Adds a column containing a constant cell in each row.
n. Column Filter: The Column Filter allows columns to be excluded from the input table.
o. Row Filter: Allows filtering of data rows by certain criteria, such as row ID, attribute value, and row number range.

p. **Naive Bayes Learner**: Creates a naive Bayes model from the given classified data. It uses ***ShearTypeClassDesc*** as target column.
q. Column Filter: The Column Filter allows columns to be excluded from the input table.
r. **Naive Bayes Predictor**: Uses the PMML naive Bayes model from the naive Bayes learner to predict the class membership of each row in the input data. It creates the predictive and probability columns for each class.

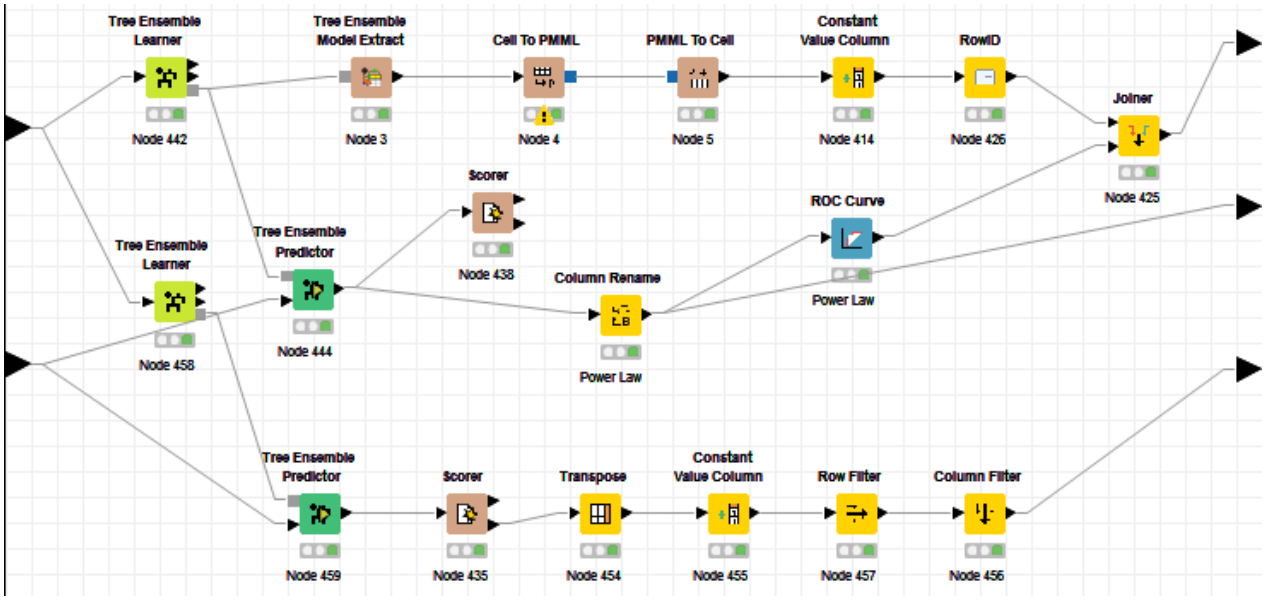
7. **Logistic Regression** contained nodes:



a. Column Filter: The Column Filter allows columns to be excluded from the input table.
b. **Regression Predictor**: Predicts the response using a regression model. It creates the predictive and probability columns for each class.
c. **Logistic Regression Learner**: Performs a multinomial logistic regression. It uses ***ShearTypeClassROCDesc*** as target column.
d. Scorer: Compares two columns by their attribute value pairs.
e. Column Rename: Enables you to rename column names or to change their types.
f. PMML To Cell: Converts the PMML Port to a table containing the PMML cell.
g. Constant Value Column: Adds a column containing a constant cell in each row.
h. ROC Curve: Shows ROC curves
i. Joiner: Joins two tables
j. RowID: Node to replace the RowID and/or to create a column with the values of the current RowID.
k. Scorer: Compares two columns by their attribute value pairs.
l. Transpose: Transposes a table by swapping rows and columns.
m. Constant Value Column: Adds a column containing a constant cell in each row.
n. Column Filter: The Column Filter allows columns to be excluded from the input table.
o. Row Filter: Allows filtering of data rows by certain criteria, such as row ID, attribute value, and row number range.

p. Column Filter: The Column Filter allows columns to be excluded from the input table.
q. **Logistic Regression Learner**: Performs a multinomial logistic regression. It uses ***ShearTypeClassDesc*** as target column.
r. **Regression Predictor**: Predicts the response using a regression model. It creates the predictive and probability columns for each class.

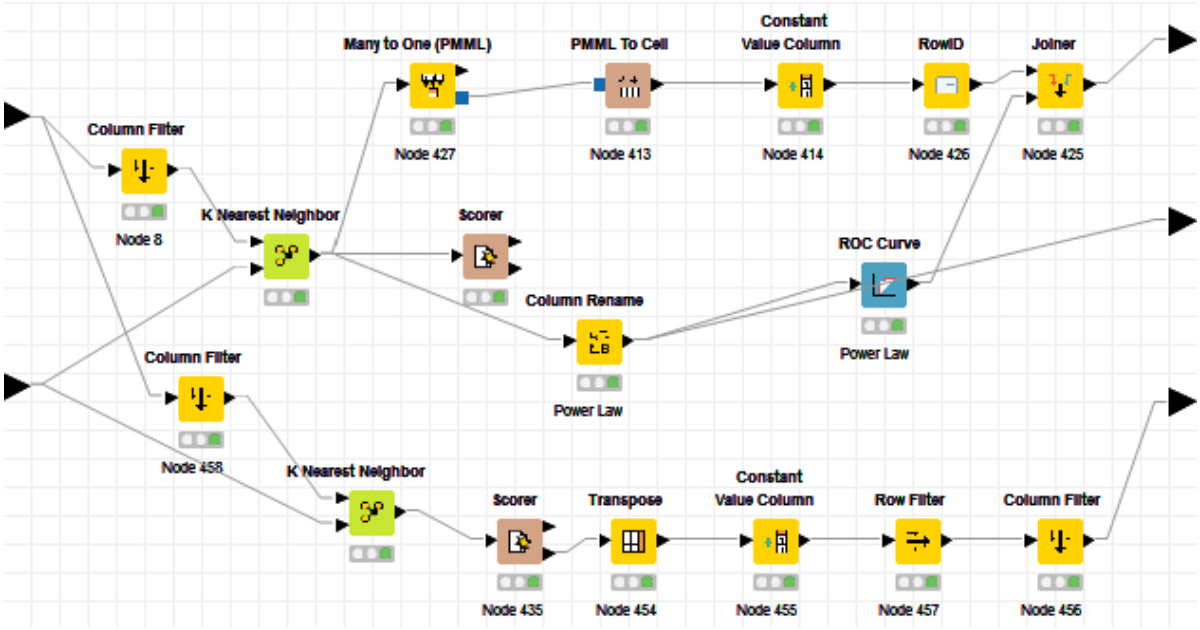
8. **Tree Ensemble** contained nodes:



a. Tree Ensemble Model Extract: Extracts individual decision trees from a tree ensemble model.
b. Cell To PMML: Converts the PMML cell in the first Row to the PMML Port.
c. PMML To Cell: Converts the PMML Port to a table containing the PMML cell.
d. Constant Value Column: Adds a column containing a constant cell in each row.
e. Joiner: Joins two tables
f. RowID: Node to replace the RowID and/or to create a column with the values of the current RowID.
g. Scorer: Compares two columns by their attribute value pairs.
h. Scorer: Compares two columns by their attribute value pairs.
i. Column Rename: Enables you to rename column names or to change their types.
j. ROC Curve: Shows ROC curves
k. **Tree Ensemble Learner**: Learns an ensemble of decision trees (such as random forest variants). It uses ***ShearTypeClassROCDesc*** as target column.
l. **Tree Ensemble Predictor**: Predicts patterns according to a majority vote in a tree ensemble model. It creates the predictive and probability columns for each class.
m. Transpose: Transposes a table by swapping rows and columns.
n. Constant Value Column: Adds a column containing a constant cell in each row.
o. Column Filter: The Column Filter allows columns to be excluded from the input table.
p. Row Filter: Allows filtering of data rows by certain criteria, such as row ID, attribute value, and row number range.
q. **Tree Ensemble Learner:** Learns an ensemble of decision trees (such as random forest variants). It uses ***ShearTypeClassDesc*** as target column.

r. **Tree Ensemble Predictor**: Predicts patterns according to a majority vote in a tree ensemble model. It creates the predictive and probability columns for each class.

9. **K Nearest Neighbor** contained nodes:



a. Column Filter: The Column Filter allows columns to be excluded from the input table.
b. **K Nearest Neighbor**: Classifies a set of test data based on the k Nearest Neighbor algorithm using the training data. It uses ***ShearTypeClassROCDesc*** as target column.
c. Scorer: Compares two columns by their attribute value pairs.
d. Column Rename: Enables you to rename column names or to change their types.
e. PMML To Cell: Converts the PMML Port to a table containing the PMML cell.
f. Constant Value Column:
g. Adds a column containing a constant cell in each row.
h. ROC Curve: Shows ROC curves
i. Joiner: Joins two tables
j. RowID: Node to replace the RowID and/or to create a column with the values of the current RowID.
k. Many to One (PMML): Transforms the values of multiple columns into a single column while generating PMML.
l. Scorer: Compares two columns by their attribute value pairs.
m. Transpose: Transposes a table by swapping rows and columns.
n. Constant Value Column: Adds a column containing a constant cell in each row.
o. Column Filter: The Column Filter allows columns to be excluded from the input table.
p. Row Filter: Allows filtering of data rows by certain criteria, such as row ID, attribute value, and row number range.
q. Column Filter: The Column Filter allows columns to be excluded from the input table.
r. **K Nearest Neighbor**: Classifies a set of test data based on the k Nearest Neighbor algorithm using the training data. It uses ***ShearTypeClassDesc*** as target column.

Note: Notice that on each model exist two learners. One is for Shear Type Class ROC (upper part) and the other for Shear Type Class (lower part). Remember that ROC means 1 for "Power Law" and 0 to otherwise.

KNIME Process Flow Explanation – Part B

Part B deals with the second goal, which is to select a predictive model to estimate the actual *shear* (α)). This is accomplished with the nodes located below in the main KNIME process flow. See next picture.

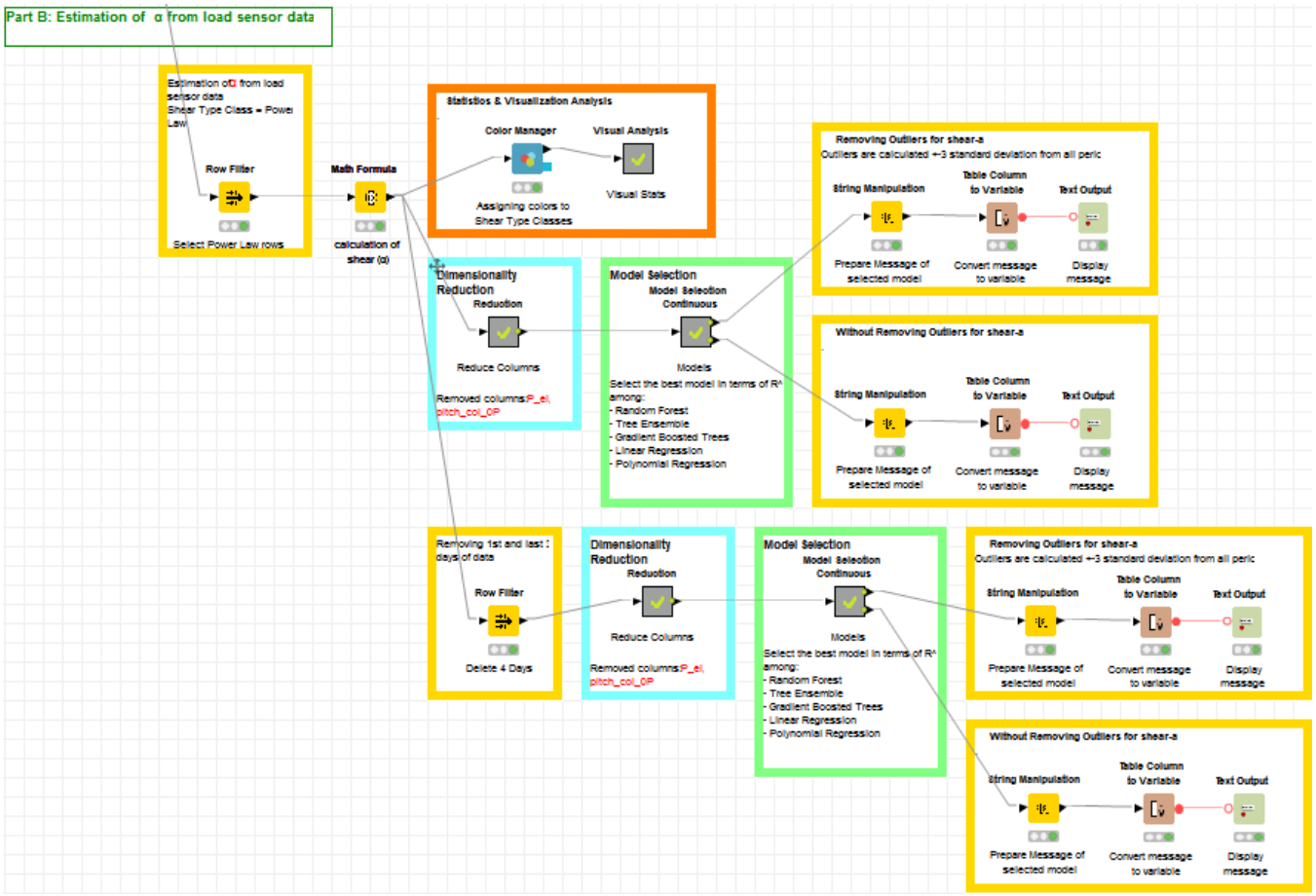


FIGURE 45 PROCESS FLOW TO ESTIMATE THE ACTUAL SHEAR- α

This process starts selecting all rows identified as "Power Law" and calculate SHEAR- α based on the formula:

$$\alpha = \frac{ln\left(\frac{V_{38m}}{V_{78.7m}}\right)}{\ln\left(\frac{38}{78.7}\right)}$$

These selection and calculation is performed with the nodes:

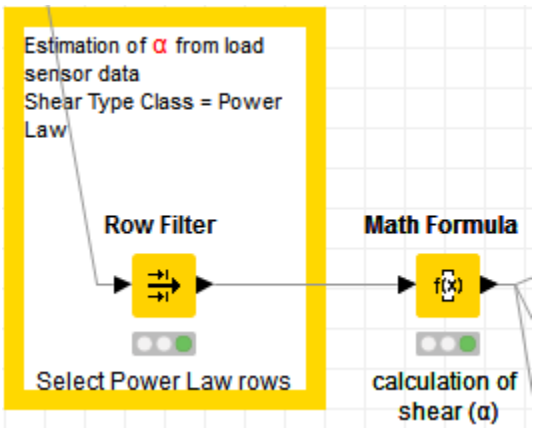


FIGURE 46 SELECTING POWER LAW ROWS AND CALCULATION OF SHEAR- α

Notice that the input of "Row Filter" node is the data prepared into the step data preparation. See subsections "Data Preparation" and "Dataset Evaluation" into "Process Flow Explanation – Part A" section.

The output of these two nodes is a table with the same structured used in Part A, but now with an additional column named ***shear_α***. which is the continuous output of what we want to predict.

Also, Part B uses the same functionality provided into the "dimensionality reduction node" described in Part A, but it is adjusted to include the column shear_a. This Part B section does not repeat the explanation for this node.

Similar to Part A, this Part B contains a special node to calculate stats for the new subset of data. See next picture:

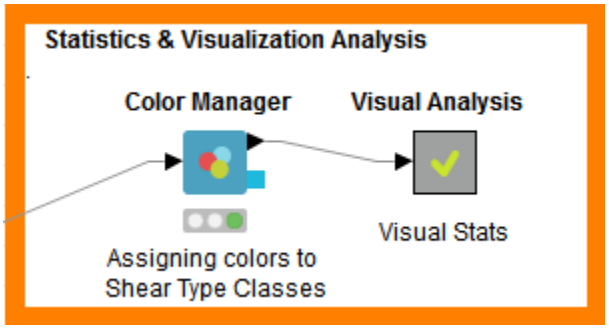


FIGURE 47 STATS AND VISUALIZATION NODES FOR SHEAR- α

Statistics and Visualization Analysis node for SHEAR- α.

In the next chart of scatter plot for **Shear- α**, notice that there are some outliers on days 19th and 26th, and almost all data in between $\mp 3\sigma$. Use "***Wind Shear Analysis.twbx***" TABLEAU file to see further details. Also, notice some outliers in the box plot for shear- α picture.

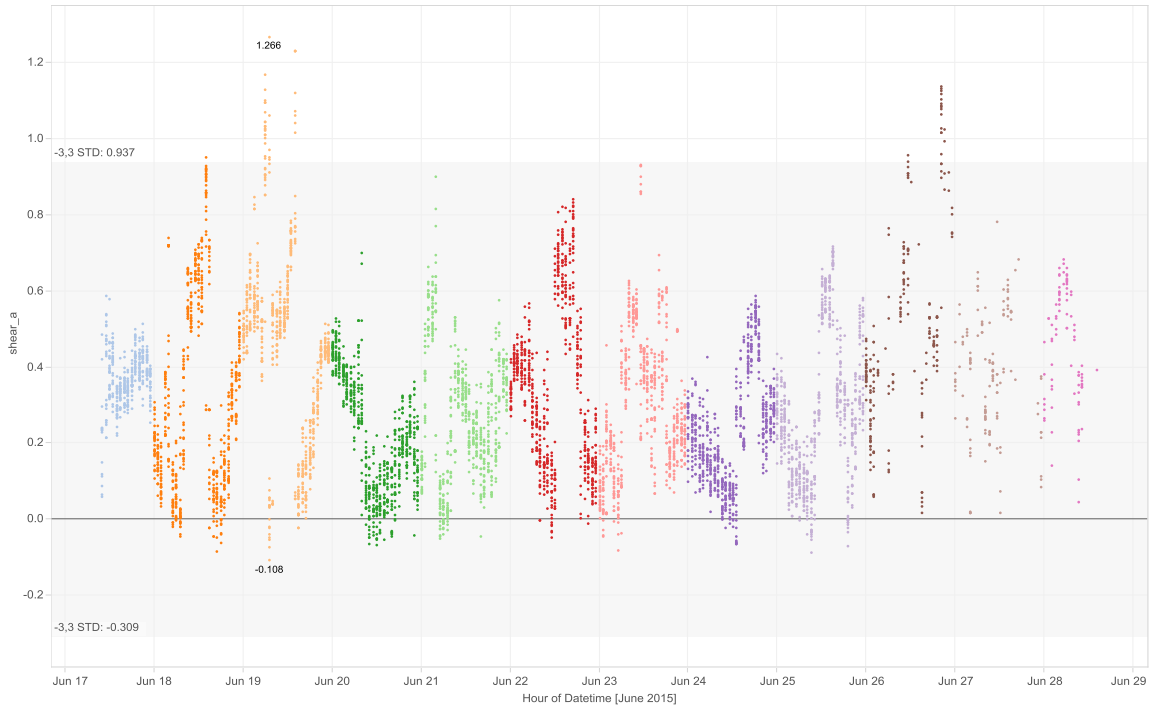


FIGURE 48 SCATTER PLOT FOR SHEAR- α

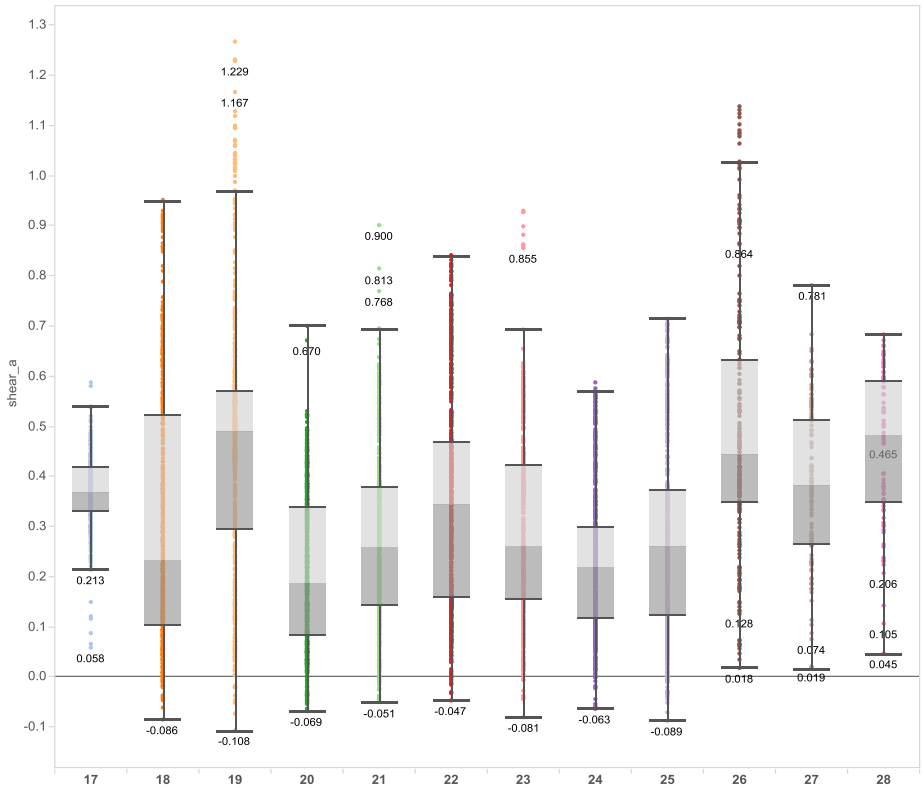


FIGURE 49 BOX PLOT FOR SHEAR- α

Column	Min	Max	Mean	Std. Dev.	Variance	Skewness	Kurtosis	Overall Sum	No. Missing	No. NaNs	No. +∞	No. -∞	Median	Row Count
m38	2.131	15.978	7.952	2.174	4.726	0.311	0.185	50,465.793	0	0	0	0	7.807	6,346
m78	2.713	17.018	9.906	2.365	5.594	0.007	-0.461	62,865.933	0	0	0	0	10.051	6,346
RPM_0P	7.992	14.618	13.507	1.492	2.227	-1.975	3.090	85,712.924	0	0	0	0	14.313	6,346
nodd_0P	-1,593.100	205.970	-605.063	297.829	88,702.106	0.041	0.719	-3,839,728.551	0	0	0	0	-690.845	6,346
nodd_3C	-267.870	430.030	27.756	63.395	4,018.891	0.747	2.757	176,141.669	0	0	0	0	20.278	6,346
nodd_3S	-183.160	415.220	77.808	75.829	5,750.109	0.686	1.857	493,772.444	0	0	0	0	70.450	6,346
pitch_d_0P	-1.586	2.433	0.404	0.764	0.583	0.789	-0.077	2,563.102	0	0	0	0	0.000	6,346
pitch_q_0P	-1.440	1.493	0.110	0.254	0.065	0.534	3.467	699.804	0	0	0	0	0.004	6,346
pitch_d_3C	-0.597	0.188	-0.069	0.087	0.008	-1.159	1.595	-438.741	0	0	0	0	-0.036	6,346
pitch_d_3S	-0.547	0.354	-0.001	0.052	0.003	-0.566	9.723	-7.675	0	0	0	0	0.000	6,346
yaw_0P	-387.520	489.130	53.531	82.616	6,825.475	0.564	1.900	339,709.398	0	0	0	0	25.548	6,346
yaw_3C	-201.440	308.760	40.402	61.429	3,773.498	0.535	1.308	256,393.467	0	0	0	0	34.387	6,346
yaw_3S	-274.140	208.730	-35.342	44.766	2,003.998	0.009	3.065	-224,282.390	0	0	0	0	-34.008	6,346
P_el	19.008	2,359.400	1,622.388	665.041	442,279.056	-0.538	-1.050	10,295,673.928	0	0	0	0	1,818.300	6,346
V_estim	3.085	16.246	9.670	2.188	4.788	-0.107	-0.425	61,368.751	0	0	0	0	9.888	6,346
pitch_col_0P	0.040	14.886	2.668	2.857	8.164	1.344	1.360	16,928.193	0	0	0	0	1.845	6,346
shear_a	-0.108	1.266	0.314	0.208	0.043	0.673	0.520	1,993.904	0	0	0	0	0.297	6,346

TABLE 11 STATISTICS OF LOAD SENSORS DATA VARIABLES (ALL DAYS) FOR SHEAR-α

Similar to PART-A, from the previous table, we can notice some important aspects for each load sensor variables:

- No NULL data in all sensor data variables.
- The skewness (symmetry in a distribution) is not zero, which means they are not having normal distribution.
- The kurtosis for some variables are bigger than 3, then the dataset for that variable has heavier tails than normal distribution.
- **NOTE:** The variances are big for some variables. Contrary to PART-A, the variance is not affected by the mix of shear types due to it only includes “Power Law” shear type.
- Notice in most of the cases, MEAN and MEDIAN are distant, which reflects distributions have skewness.

What will be the behavior of these statistics if we ***remove the 4 days*** (days 17, 26, 27, 28) where data seems to be not completed for Power Law shear type? In general, it is the same behavior, some stats become better and some others become worst. See below table.

Column	Min	Max	Mean	Std. Dev.	Variance	Skewness	Kurtosis	Overall Sum	No. Missing	No. NaNs	No. +∞	No. -∞	Median	Row Count
m38	2.341	15.978	8.021	2.185	4.775	0.335	0.077	44,149.660	0	0	0	0	7.822	5,504
m78	2.713	17.007	9.889	2.359	5.564	0.047	-0.418	54,428.248	0	0	0	0	10.003	5,504
RPM_0P	7.999	14.567	13.516	1.488	2.215	-1.981	3.063	74,393.437	0	0	0	0	14.313	5,504
nodd_0P	-1,593.100	205.970	-599.019	306.249	93,788.680	0.003	0.609	-3,296,998.758	0	0	0	0	-690.270	5,504
nodd_3C	-267.870	360.120	22.802	57.561	3,313.214	0.402	2.401	125,503.385	0	0	0	0	18.813	5,504
nodd_3S	-183.160	404.190	75.036	73.279	5,369.755	0.533	1.765	412,996.857	0	0	0	0	69.830	5,504
pitch_d_0P	-1.586	2.433	0.352	0.754	0.569	0.873	0.139	1,939.887	0	0	0	0	0.000	5,504
pitch_q_0P	-1.440	1.493	0.094	0.251	0.063	0.484	3.749	516.406	0	0	0	0	0.002	5,504
pitch_d_3C	-0.597	0.188	-0.067	0.084	0.007	-1.051	1.209	-367.322	0	0	0	0	-0.034	5,504
pitch_d_3S	-0.420	0.354	-0.001	0.052	0.003	-0.473	7.736	-6.221	0	0	0	0	0.000	5,504
yaw_0P	-387.520	489.130	54.286	84.134	7,078.552	0.511	1.901	298,787.409	0	0	0	0	27.479	5,504
yaw_3C	-201.440	308.760	37.007	58.495	3,421.618	0.429	1.441	203,684.184	0	0	0	0	32.730	5,504
yaw_3S	-274.140	208.730	-33.111	43.503	1,892.474	0.097	3.431	-182,241.292	0	0	0	0	-32.868	5,504
P_el	19.008	2,359.400	1,616.187	657.633	432,481.661	-0.533	-1.019	8,895,494.615	0	0	0	0	1,792.650	5,504
V_estim	3.085	16.170	9.665	2.185	4.775	-0.057	-0.387	53,193.503	0	0	0	0	9.840	5,504
pitch_col_0P	0.040	14.541	2.654	2.891	8.357	1.399	1.477	14,609.657	0	0	0	0	1.782	5,504
shear_a	-0.108	1.266	0.299	0.208	0.043	0.702	0.329	1,647.023	0	0	0	0	0.268	5,504

TABLE 12 STATISTICS OF LOAD SENSORS DATA VARIABLES (WITHOUT DAYS 17, 26, 27, 28) FOR SHEAR- α

The stats for both tables above show similar results, so seems there is no too much difference if days 17, 26, 27 and 28 are included. Nevertheless, both scenarios are included in the model selection processes into KNIME.

The next picture shows the histogram for SHEAR- α. Notice there are around 363 records close to ZERO, that could affect the performance of the model evaluation.

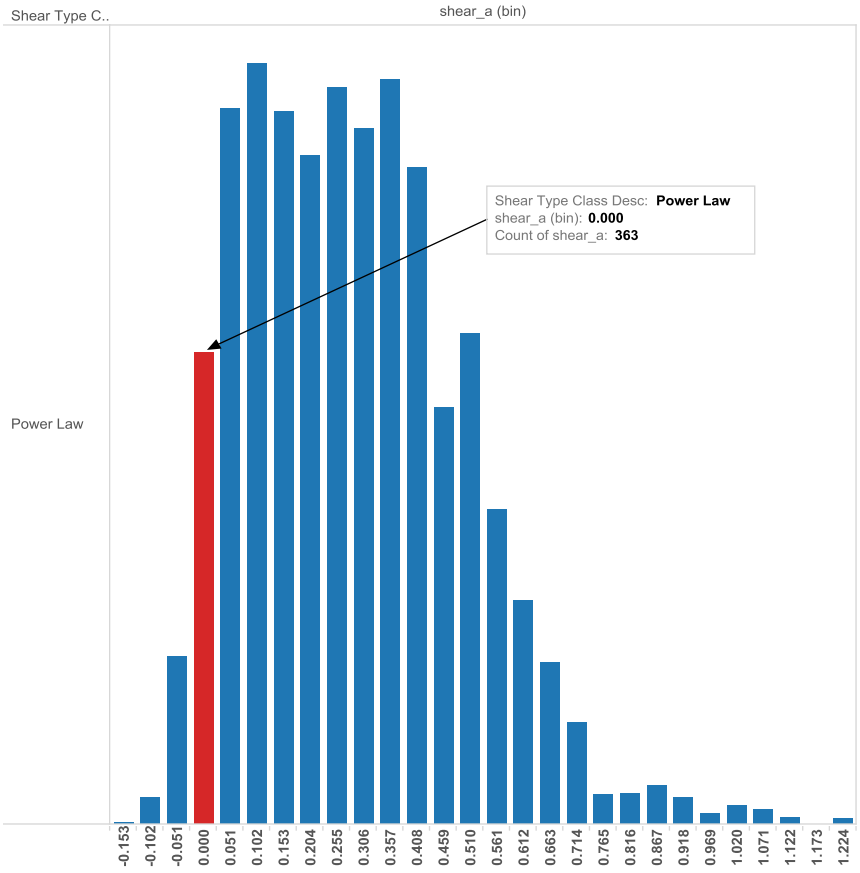


FIGURE 50 HISTOGRAMS FOR SHEAR- α

How much are correlated the variables now including SHEAR- α? Similar to PART A, the follow charts show if there is a strong or weak correlation between variables. Remember, the color range varies from dark red (strong negative correlation), over white (no correlation), to dark blue (strong positive correlation).

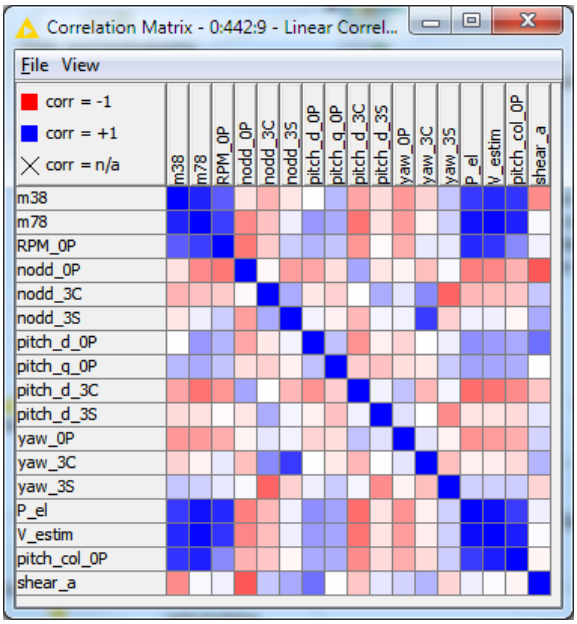


FIGURE 51 CORRELATION INCLUDING SHEAR- α

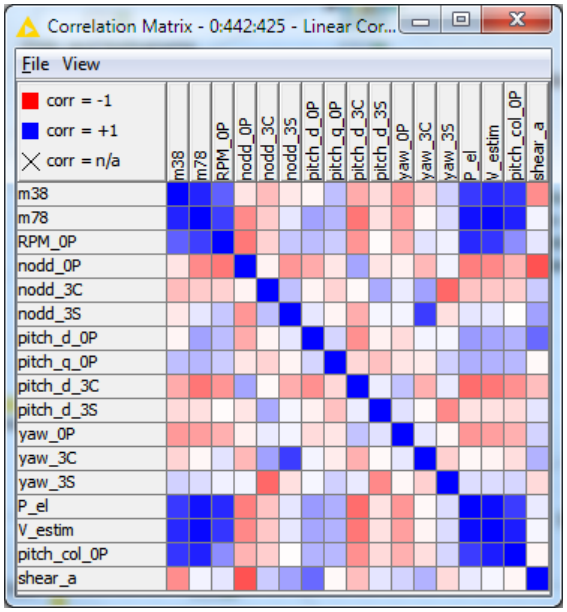


FIGURE 52 CORRELATION INCLUDING SHEAR- α (WITHOUT 4 DAYS)

.	m38	m78	RPM_0P	nodd_0P	nodd_3C	nodd_3S	pitch_d_0P	pitch_q_0P	pitch_d_3C	pitch_d_3S	yaw_0P	yaw_3C	yaw_3S	P_el	V_estim	pitch_col_0P	shear_a
m38	1	0.865	0.639	-0.118	-0.295	-0.098	0.009	0.280	-0.357	-0.137	-0.405	-0.177	0.228	0.782	0.847	0.794	-0.455
m78	0.865	1	0.766	-0.464	-0.238	0.065	0.407	0.335	-0.549	-0.116	-0.390	-0.051	0.181	0.937	0.975	0.877	0.027
RPM_0P	0.639	0.766	1	-0.529	-0.207	0.201	0.292	0.233	-0.417	-0.020	-0.327	0.082	0.090	0.848	0.799	0.459	0.059
nodd_0P	-0.118	-0.464	-0.529	1	-0.026	-0.382	-0.351	-0.130	0.348	-0.100	-0.045	-0.251	0.029	-0.507	-0.473	-0.306	-0.659
nodd_3C	-0.295	-0.238	-0.207	-0.026	1	0.327	-0.091	-0.185	-0.010	0.326	0.096	0.460	-0.613	-0.277	-0.258	-0.221	0.220
nodd_3S	-0.098	0.065	0.201	-0.382	0.327	1	0.049	-0.055	-0.284	0.040	0.048	0.778	-0.184	0.060	0.061	-0.034	0.339
pitch_d_0P	0.009	0.407	0.292	-0.351	-0.091	0.049	1	0.242	-0.429	-0.063	-0.174	-0.004	0.067	0.444	0.396	0.339	0.566
pitch_q_0P	0.280	0.335	0.233	-0.130	-0.185	-0.055	0.242	1	-0.196	-0.242	-0.130	-0.088	0.204	0.365	0.349	0.319	0.004
pitch_d_3C	-0.357	-0.549	-0.417	0.348	-0.010	-0.284	-0.429	-0.196	1	0.050	0.238	-0.281	0.037	-0.574	-0.539	-0.450	-0.224
pitch_d_3S	-0.137	-0.116	-0.020	-0.100	0.326	0.040	-0.063	-0.242	0.050	1	0.121	-0.002	-0.452	-0.112	-0.115	-0.144	0.096
yaw_0P	-0.405	-0.390	-0.327	-0.045	0.096	0.048	-0.174	-0.130	0.238	0.121	1	0.101	-0.053	-0.422	-0.390	-0.320	0.173
yaw_3C	-0.177	-0.051	0.082	-0.251	0.460	0.778	-0.004	-0.088	-0.281	-0.002	0.101	1	-0.251	-0.064	-0.069	-0.144	0.292
yaw_3S	0.228	0.181	0.090	0.029	-0.613	-0.184	0.067	0.204	0.037	-0.452	-0.053	-0.251	1	0.183	0.183	0.197	-0.168
P_el	0.782	0.937	0.848	-0.507	-0.277	0.060	0.444	0.365	-0.574	-0.112	-0.422	-0.064	0.183	1	0.961	0.768	0.063
V_estim	0.847	0.975	0.799	-0.473	-0.258	0.061	0.396	0.349	-0.539	-0.115	-0.390	-0.069	0.183	0.961	1	0.891	0.018
pitch_col_0P	0.794	0.877	0.459	-0.306	-0.221	-0.034	0.339	0.319	-0.450	-0.144	-0.320	-0.144	0.197	0.768	0.891	1	-0.040
shear_a	-0.455	0.027	0.059	-0.659	0.220	0.339	0.566	0.004	-0.224	0.096	0.173	0.292	-0.168	0.063	0.018	-0.040	1

TABLE 13 CORRELATION VALUES TABLE FOR SHEAR- α (ALL DAYS).

.	m38	m78	RPM_0P	nodd_0P	nodd_3C	nodd_3S	pitch_d_0P	pitch_q_0P	pitch_d_3C	pitch_d_3S	yaw_0P	yaw_3C	yaw_3S	P_el	V_estim	pitch_col_0P	shear_a
m38	1	0.862	0.622	-0.105	-0.264	-0.089	-0.040	0.253	-0.331	-0.144	-0.401	-0.168	0.182	0.777	0.842	0.793	-0.447
m78	0.862	1	0.757	-0.459	-0.205	0.094	0.364	0.283	-0.536	-0.120	-0.380	-0.029	0.137	0.934	0.973	0.875	0.045
RPM_0P	0.622	0.757	1	-0.531	-0.177	0.220	0.257	0.199	-0.415	-0.017	-0.310	0.110	0.055	0.845	0.791	0.447	0.097
nodd_0P	-0.105	-0.459	-0.531	1	-0.045	-0.411	-0.330	-0.097	0.351	-0.104	-0.058	-0.278	0.049	-0.506	-0.469	-0.297	-0.678
nodd_3C	-0.264	-0.205	-0.177	-0.045	1	0.247	-0.043	-0.173	-0.025	0.334	0.080	0.368	-0.594	-0.234	-0.221	-0.193	0.205
nodd_3S	-0.089	0.094	0.220	-0.411	0.247	1	0.095	-0.045	-0.321	0.034	0.041	0.763	-0.121	0.100	0.091	-0.008	0.370
pitch_d_0P	-0.040	0.364	0.257	-0.330	-0.043	0.095	1	0.165	-0.436	-0.054	-0.142	0.044	0.034	0.394	0.351	0.299	0.589
pitch_q_0P	0.253	0.283	0.199	-0.097	-0.173	-0.045	0.165	1	-0.155	-0.242	-0.100	-0.075	0.185	0.313	0.304	0.279	-0.023
pitch_d_3C	-0.331	-0.536	-0.415	0.351	-0.025	-0.321	-0.436	-0.155	1	0.070	0.237	-0.310	0.081	-0.574	-0.533	-0.437	-0.255
pitch_d_3S	-0.144	-0.120	-0.017	-0.104	0.334	0.034	-0.054	-0.242	0.070	1	0.119	-0.023	-0.464	-0.115	-0.118	-0.147	0.102
yaw_0P	-0.401	-0.380	-0.310	-0.058	0.080	0.041	-0.142	-0.100	0.237	0.119	1	0.092	-0.030	-0.410	-0.377	-0.311	0.171
yaw_3C	-0.168	-0.029	0.110	-0.278	0.368	0.763	0.044	-0.075	-0.310	-0.023	0.092	1	-0.186	-0.028	-0.044	-0.130	0.302
yaw_3S	0.182	0.137	0.055	0.049	-0.594	-0.121	0.034	0.185	0.081	-0.464	-0.030	-0.186	1	0.135	0.138	0.159	-0.147
P_el	0.777	0.934	0.845	-0.506	-0.234	0.100	0.394	0.313	-0.574	-0.115	-0.410	-0.028	0.135	1	0.959	0.764	0.083
V_estim	0.842	0.973	0.791	-0.469	-0.221	0.091	0.351	0.304	-0.533	-0.118	-0.377	-0.044	0.138	0.959	1	0.891	0.037
pitch_col_0P	0.793	0.875	0.447	-0.297	-0.193	-0.008	0.299	0.279	-0.437	-0.147	-0.311	-0.130	0.159	0.764	0.891	1	-0.035
shear_a	-0.447	0.045	0.097	-0.678	0.205	0.370	0.589	-0.023	-0.255	0.102	0.171	0.302	-0.147	0.083	0.037	-0.035	1

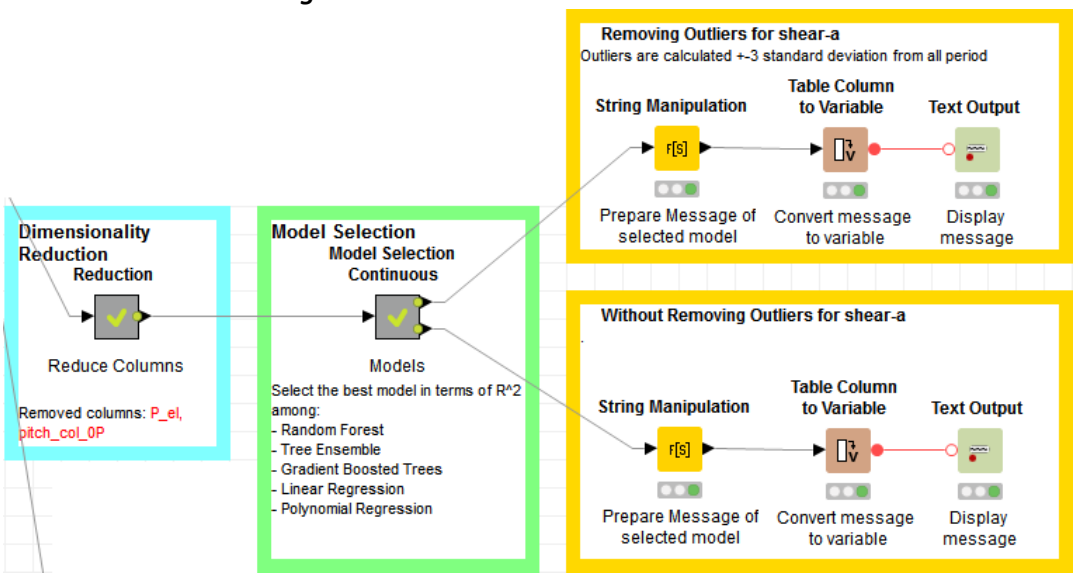
TABLE 14 CORRELATION VALUES TABLE FOR SHEAR- α (WITHOUT 4 DAYS).

Notice that there is a strong negative correlation between shear_a and nodd_0P that could affect the performance of the model selection.

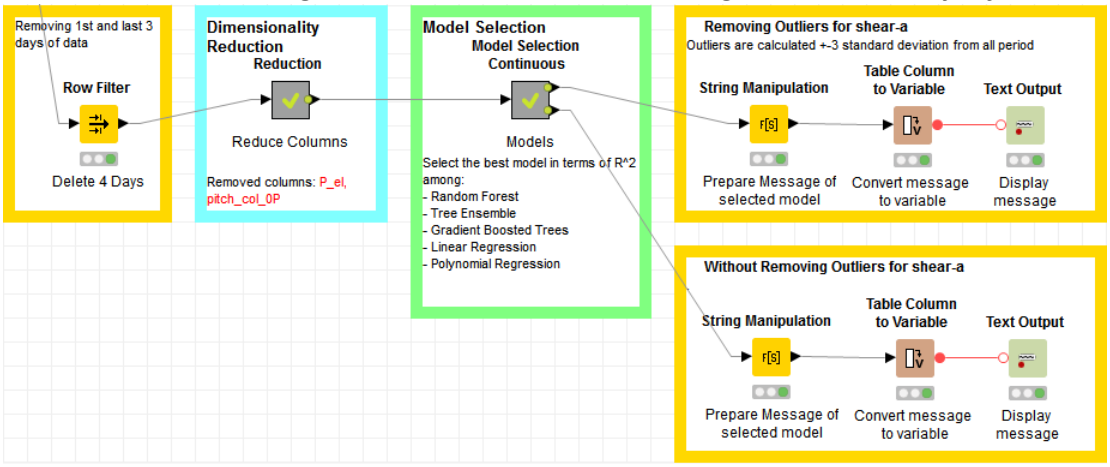
Model Selection node for continuous output.

Similar to Part A, in Part B there is a Model Selection Continuous node which is the core for the classification or prediction model selection. This node identifies the best model depending the scenario under analysis. The scenarios are:

A. Test the models ***removing correlated columns***.



B. Test the models ***removing correlated columns***, and ***removing 1st and last 3 days of data***.



Note: the nodes "***String Manipulation***", "***Table Column to Variable***" and "***Text Output***" are used to display the message containing the model selected and the its value. Here is an example of this message:

Label
Selected Model: Tree Ensemble with R^2 = 0.8990976505950915.

Model Selection node in details for continuous output

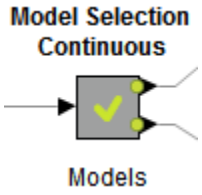


FIGURE 53 MODEL SELECTION NODE FOR CONTINUOUS

This node selects the best model in terms of ***R-Squared*** among following models (why they? – see appendix):

- Random Forest
- Tree Ensemble
- Gradient Boosted Trees
- Linear Regression
- Polynomial Regression

Different to Part A, notice this node has one input and two outputs (this node in Part A has 1 input and 3 outputs):

- Input: It takes the dataset to be analyzed according scenarios described above. This data set removes those correlated columns before to be used by this Model Selection Continuous model.
- Output 1: The best model evaluation in terms of R-Squared, but removing Outliers for ***shear-a***.
  Note: Each model evaluates the column ***shear_a*** against its ***Prediction (shear_a)*** column.
- Output 2: is the same as Output 1, but without removing Outliers for ***shear-a***.

If we open the Model Selection Continuous node, we can see the next nodes inside.

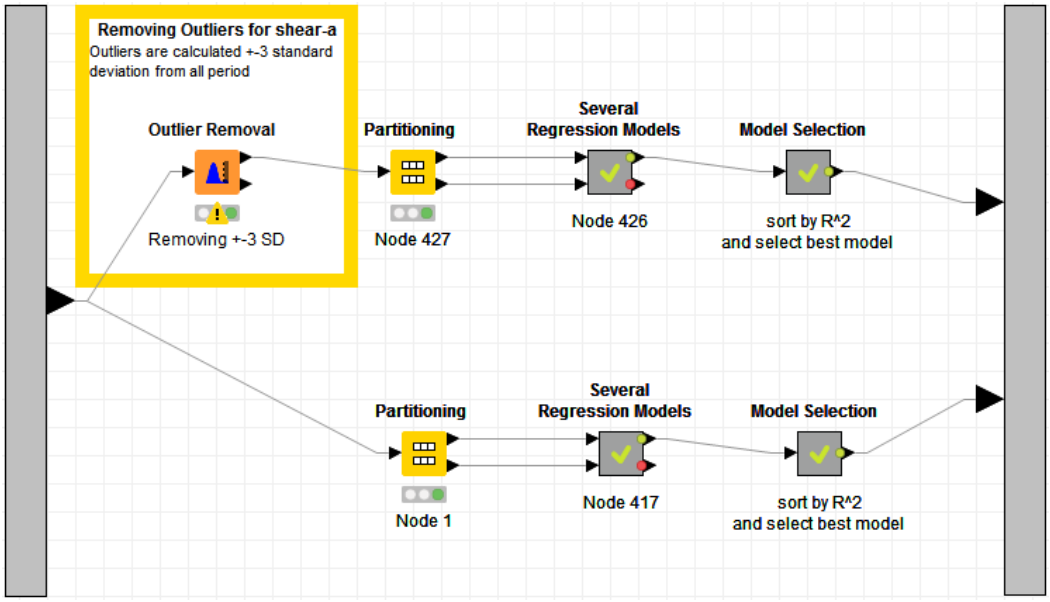


FIGURE 54 MODEL SELECTION CONTINUOUS NODE IN DETAILS

Notice that upper flow looks quite similar to lower flow. They are similar. The only difference between them is that the upper flow start removing outliers of ***shear_a***. The values outside of $\mp 3\sigma$ are removed. This outlier removal is performed by:

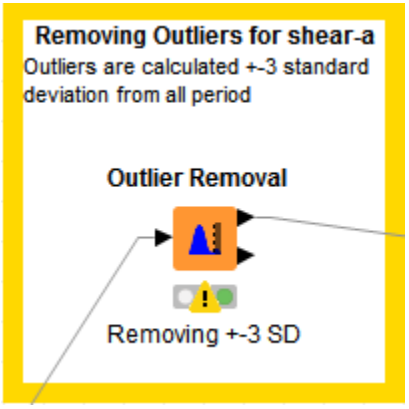


FIGURE 55 OUTLIER REMOVAL NODE

The node containing all models that compete are into:

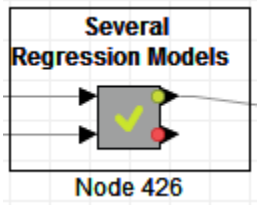


FIGURE 56 SEVERAL EGRESSION MODELS NODE

Open the Several Regression Models to see details. See next picture.

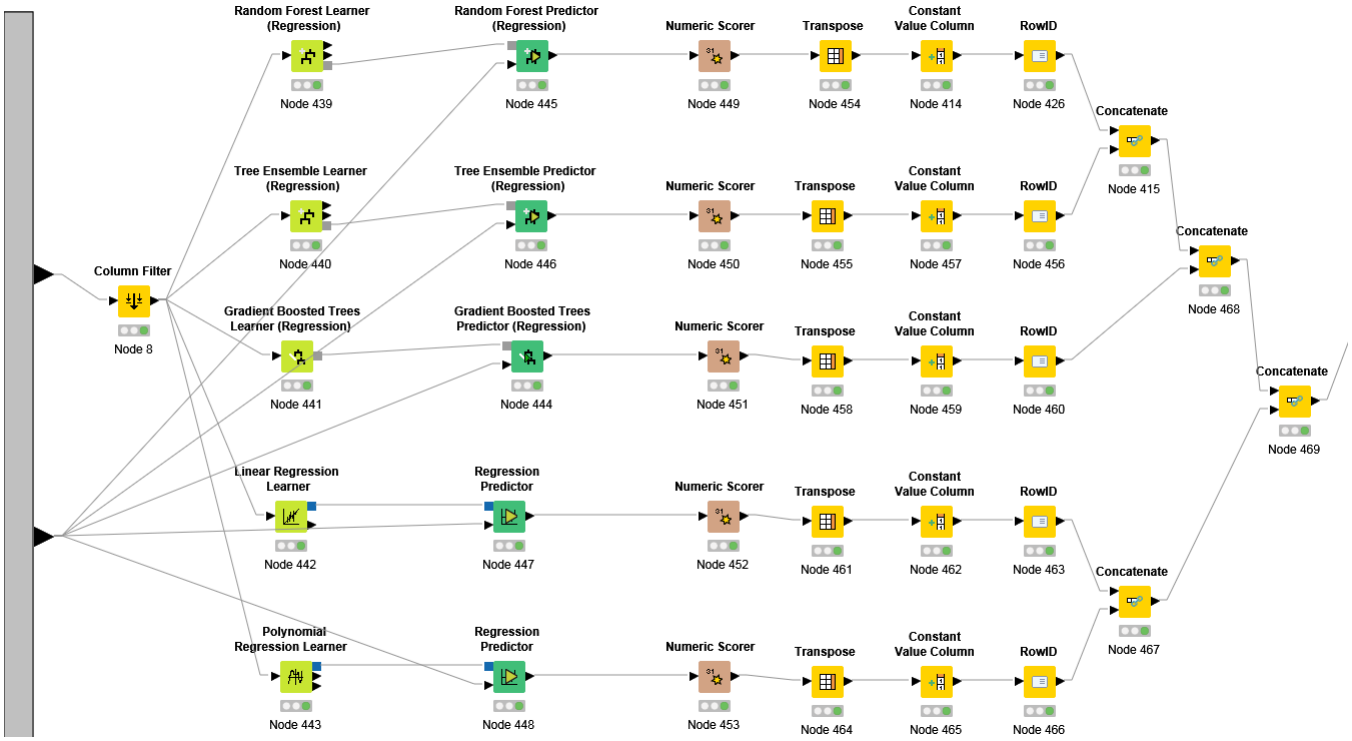
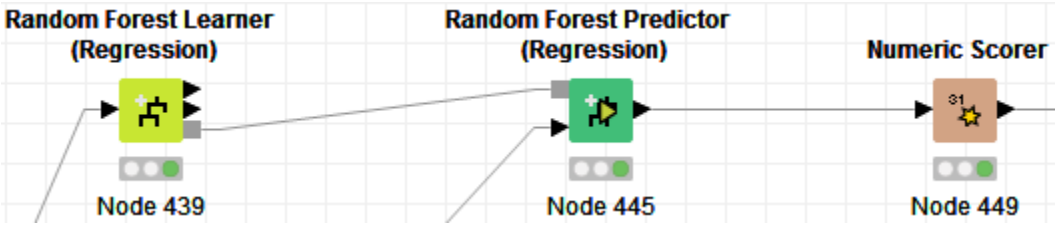


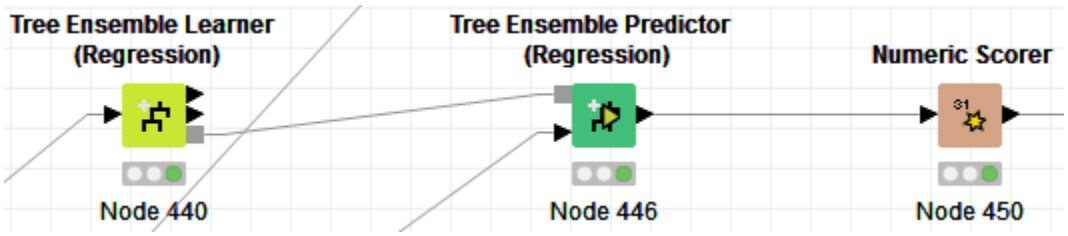
FIGURE 57 SEVERAL EGRESSION MODELS NODE IN DETAILS

Steps:

1. Column Filter: The Column Filter allows columns to be excluded from the input table.
2. Random Forest



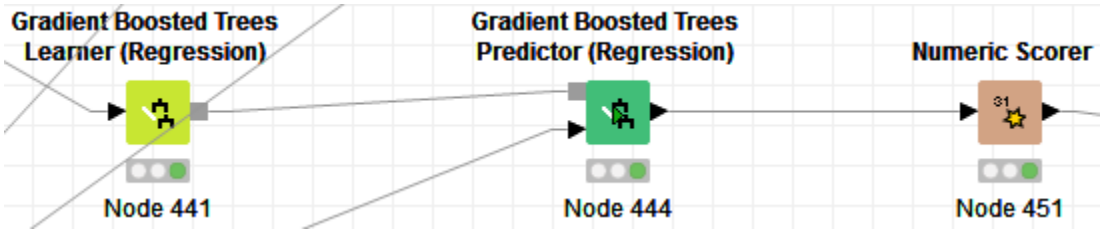
   a. Random Forest Learner (Regression): Learns a random forest for regression.
   b. Random Forest Predictor (Regression): Applies regression from a random forest model by using the mean of the individual predictions.
   c. Numeric Scorer: Computes certain statistics between a numeric column's values and predicted values.
3. Tree Ensemble



   a. Tree Ensemble Learner (Regression): Learns an ensemble of regression trees.

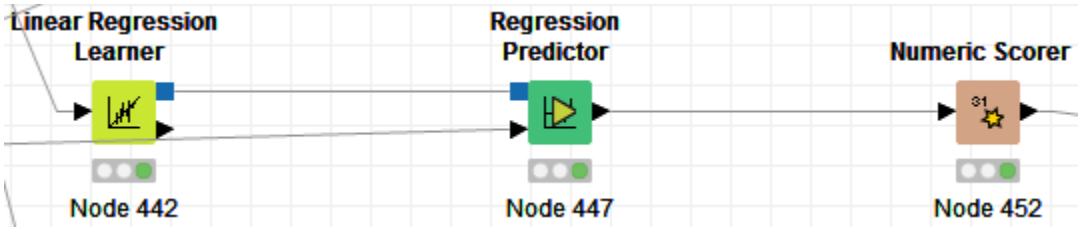
b. Tree Ensemble Predictor (Regression): Applies regression from a tree ensemble model by using the mean of the individual predictions.
c. Numeric Scorer: Computes certain statistics between a numeric column's values and predicted values.

4. Gradient Boosted Trees



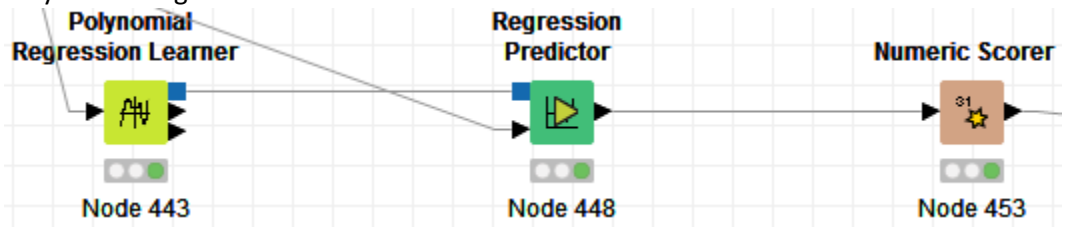
a. Gradient Boosted Trees Learner (Regression): Learns a Gradient Boosted Trees model.
b. Gradient Boosted Trees Predictor (Regression): Applies regression from a Gradient Boosted Trees model.
c. Numeric Scorer: Computes certain statistics between a numeric column's values and predicted values.

5. Linear Regression



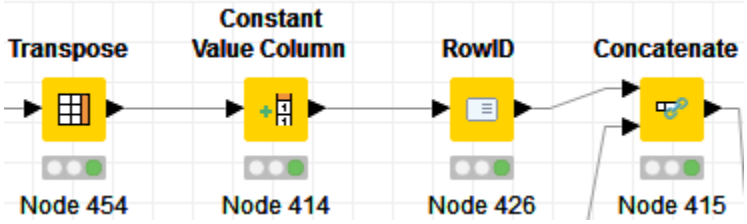
a. Linear Regression Learner: Performs a multivariate linear regression.
b. Regression Predictor: Predicts the response using a regression model.
c. Numeric Scorer: Computes certain statistics between a numeric column's values and predicted values.

6. Polynomial Regression



a. Polynomial Regression Learner: Learner that builds a polynomial regression model from the input data
b. Regression Predictor: Predicts the response using a regression model.
c. Numeric Scorer: Computes certain statistics between a numeric column's values and predicted values.

7. Auxiliary nodes



   a. Transpose: Transposes a table by swapping rows and columns.
   b. Constant Value Column: Adds a column containing a constant cell in each row.
   c. RowID: Node to replace the RowID and/or to create a column with the values of the current RowID.
   d. Concatenate: Concatenates two tables row-wise.

KNIME Process Flow Explanation – Part C

Part C goal

Part C: Estimation of **Shear** when **speed profile does not follow a power law**. For speed profiles labeled as 1, 2, and 3, how might you create a model to estimate these profiles?

What to do

Remember that **shear (α)** can be calculated by using wind speeds at altitudes of 38 m and 78.7 m when the speed profile follows a power law (see following equations).

$$V_{38m} = V_{78.7m}\left(\frac{38}{78.7}\right)^{\alpha}$$

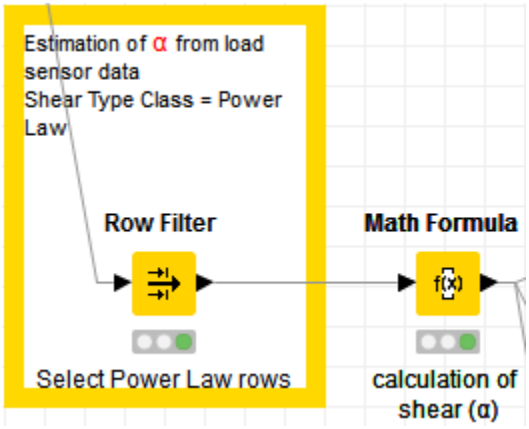
$$\alpha = \frac{ln\left(\frac{V_{38m}}{V_{78.7m}}\right)}{\ln\left(\frac{38}{78.7}\right)}$$

However, the "*Wind Shear Case Study"* document does not mention if the same equation can be used for speed profiles that does not follow a power law.

If we want to deal with this Part C goal similar as we did for Part B, we need to answer some questions, or to do some assumptions.

Points to consider:

- Ask to SME (subject matter experts) if the same equation (just described above) can be used for speed profiles that do not follow a power law; or, if there is a specific equation for each speed profile when ShearTypeClass is: 1 = LLJ, 2 = Flat, and 3 = Others.
    - If answer is YES, que can adjust the process followed in PART B accordingly.
    - If NO, define the new equations for types 1, 2 and 3; then adjust the process followed in PART B using new equations accordingly. In special these nodes:



- As reviewed in section "***Good to know***" (above in this document). There are two well-defined factors affecting wind speed: *environmental factors*, and *artificial factors*. The *friction coefficient* α is set empirically and the equation can be used to adjust the data reasonably well in the range of 10 up to 100-150 metres.
    - In this case we have to find out a way to extrapolate the wind speed data and identify which equations can be used for types 1, 2 and 3; then adjust the process followed in PART B using new equations accordingly.
- As reviewed in section "Exploration of data" (above in this document). There are few data for the types 1, 2 and 3 in the dataset provided (the total number of rows for these 3 types is 1,700). Also, the data collected for these 3 types, seem that only 3 days have representative data for these types (1,436 records of 1,700 were collected in 3 days for these 3 types).
    - In this case, additionally to identify which equation to apply, it is recommended to collect more data for types 1, 2 and 3
- In general, to deal with PART C, more explanation is required from SMEs.

Analysis

This section shows the results of the tested models.

Results for Part A

The solutions were divided in different scenarios (see "KNIME Process Flow Explanation – Part A" section). Likewise, the results are presented for each scenario.

Each model is evaluated based on:

1. Its AuC (Area Under Curve), which takes the shear type "Power Law" as the positive class value (True Positives).
   Interpreting an AUC of 0.7 for example means that a randomly selected case from the group with the target equals 1(Power Law) has a score larger than that for a randomly chosen case from the group with the target equals 0 (No Power Law) in 70% of the time. Next table shows accuracy classification by AUC.
   if the classifier is very good, the true positive rate will increase quickly and the area under the curve will be close to 1. If the classifier is no better than random guessing, the true positive rate will increase linearly with the false positive rate and the area under the curve will be around 0.5.
2. Its Recall, Precision, Sensitivity, Specificity, and F-measure; which are evaluated under two scenarios:
   a. Shear type values of "Power Law" and "No Power Law" – binary classification, and
   b. Shear type values of "Power Law", "Flat", "LLJ", and "Others" – nominal classification.

AUC Range	Classification
0.9 < AUC < 1.0	Excellent
0.8 < AUC < 0.9	Good
0.7 < AUC < 0.8	Worthless
0.6 < AUC < 0.7	Not good

TABLE 15 ACCURACY CLASSIFICATION BY AUC FOR A DIAGNOSTIC TEST[6]

[6] Sensitivity, Specificity, Accuracy, Associated Confidence Interval and ROC. http://www.lexjansen.com/nesug/nesug10/hl/hl07.pdf

What does contain the result table?

- Left side shows how many true positives (light blue in diagonal) and true negatives.
- The Area Under Curve is shown in Orange/Gold color.
- Right next, it is a summary table, and the accuracy in highlighted in light green.
- Right table shows Agreement, By Chance, Confusion Matrix (True Positives, False Positives, False Negatives, True Negatives), Recall, Precision, Sensitivity, Specificity, and F-measure.

Results of testing the models without removing correlated columns.

Random Forest

Random Forest				
	Power Law	No Power Law		
Power Law	1,251	28		
No Power Law	99	232		
Area Under Curve:		**0.928**		

Total # of subjects:	**1,610**
Correct classified:	**1,483**
Wrong classified:	**127**
Accuracy:	**92.112%**
Error:	**7.888%**
Cohen's kappa (k):	**0.738**

				Type II error	Type I error						
Agreement	**By Chance**	**TruePositives**	**FalsePositives**	**FalseNegatives**	**TrueNegatives**	**Recall**	**Precision**	**Sensitivity**	**Specifity**	**F-measure**
1,251	1,072	1,251	99	28	232	0.978	0.927	0.978	0.701	0.952
232	53	232	28	99	1,251	0.701	0.892	0.701	0.978	0.785
-	-	-	-	-	-	-	-	-	-	-
-	-	-	-	-	-	-	-	-	-	-
1,483	**1,126**									

Random Forest considering shear types 0,1,2,3				
	Power Law	Flat	LLJ	Others
Power Law	1,259	4	15	1
Flat	48	36	0	0
LLJ	37	0	190	1
Others	14	1	1	3

Total # of subjects:	**1,610**
Correct classified:	**1,488**
Wrong classified:	**122**
Accuracy:	**92.422%**
Error:	**7.578%**
Cohen's kappa (k):	**0.756**

				Type II error	Type I error						
Agreement	**By Chance**	**TruePositives**	**FalsePositives**	**FalseNegatives**	**TrueNegatives**	**Recall**	**Precision**	**Sensitivity**	**Specifity**	**F-measure**
1,259	1,079	1,259	99	20	232	0.984	0.927	0.984	0.701	0.955
36	2	36	5	48	1,521	0.429	0.878	0.429	0.997	0.576
190	29	190	16	38	1,366	0.833	0.922	0.833	0.988	0.876
3	0	3	2	16	1,469	0.158	0.600	0.158	0.999	0.250
1,488	**1,110**									

Decision Tree (using R)

Decision Tree - R				
	Power Law	No Power Law		
Power Law	1,241	38		
No Power Law	173	158		
Area Under Curve:		**0.756**		

Total # of subjects:	**1,610**
Correct classified:	**1,399**
Wrong classified:	**211**
Accuracy:	**86.894%**
Error:	**13.106%**
Cohen's kappa (k):	**0.527**

				Type II error	Type I error						
Agreement	**By Chance**	**TruePositives**	**FalsePositives**	**FalseNegatives**	**TrueNegatives**	**Recall**	**Precision**	**Sensitivity**	**Specifity**	**F-measure**
1,241	1,123	1,241	173	38	158	0.970	0.878	0.970	0.477	0.922
158	40	158	38	173	1,241	0.477	0.806	0.477	0.970	0.600
-	-	-	-	-	-	-	-	-	-	-
-	-	-	-	-	-	-	-	-	-	-
1,399	**1,164**									

Decision Tree - R considering shear types 0,1,2,3				
	Power Law	Flat	LLJ	Others
Power Law	1,176	33	58	12
Flat	51	32	0	1
LLJ	37	0	190	1
Others	13	1	2	3

Total # of subjects:	1,610
Correct classified:	1,401
Wrong classified:	209
Accuracy:	87.019%
Error:	12.981%
Cohen's kappa (k):	0.624

Agreement	By Chance	TruePositives	Type II error FalsePositives	Type I error FalseNegatives	TrueNegatives	Recall	Precision	Sensitivity	Specifity	F-measure
1,176	1,014	1,176	101	103	230	0.919	0.921	0.919	0.695	0.920
32	3	32	34	52	1,492	0.381	0.485	0.381	0.978	0.427
190	35	190	60	38	1,322	0.833	0.760	0.833	0.957	0.795
3	0	3	14	16	1,369	0.158	0.176	0.158	0.990	0.167
1,401	1,054									

Naive Bayes

Naïve Bayes				
	Power Law	No Power Law		
Power Law	910	369		
No Power Law	92	239		
Area Under Curve:		0.780		

Total # of subjects:	1,610
Correct classified:	1,149
Wrong classified:	461
Accuracy:	71.366%
Error:	28.634%
Cohen's kappa (k):	0.331

Agreement	By Chance	TruePositives	Type II error FalsePositives	Type I error FalseNegatives	TrueNegatives	Recall	Precision	Sensitivity	Specifity	F-measure
910	796	910	92	369	239	0.711	0.908	0.711	0.722	0.798
239	125	239	369	92	910	0.722	0.393	0.722	0.711	0.509
-	-	-	-	-	-	-	-	-	-	-
-	-	-	-	-	-	-	-	-	-	-
1,149	921									

Naïve Bayes considering shear types 0,1,2,3				
	Power Law	Flat	LLJ	Others
Power Law	1,044	10	201	24
Flat	61	10	13	0
LLJ	29	1	174	24
Others	9	1	5	4

Total # of subjects:	1,610
Correct classified:	1,232
Wrong classified:	378
Accuracy:	76.522%
Error:	23.478%
Cohen's kappa (k):	0.414

Agreement	By Chance	TruePositives	Type II error FalsePositives	Type I error FalseNegatives	TrueNegatives	Recall	Precision	Sensitivity	Specifity	F-measure
1,044	908	1,044	99	235	232	0.816	0.913	0.816	0.701	0.862
10	1	10	12	74	1,514	0.119	0.455	0.119	0.992	0.189
174	56	174	219	54	1,163	0.763	0.443	0.763	0.842	0.560
4	1	4	48	15	1,164	0.211	0.077	0.211	0.960	0.113
1,232	965									

Logistic Regression

Logistic Regression				
	Power Law	No Power Law		
Power Law	1,246	33		
No Power Law	216	115		
Area Under Curve:		0.782		

Total # of subjects:	1,610
Correct classified:	1,361
Wrong classified:	249
Accuracy:	84.534%
Error:	15.466%
Cohen's kappa (k):	0.405

Agreement	By Chance	TruePositives	Type II error FalsePositives	Type I error FalseNegatives	TrueNegatives	Recall	Precision	Sensitivity	Specifity	F-measure
1,246	1,161	1,246	216	33	115	0.974	0.852	0.974	0.347	0.909
115	30	115	33	216	1,246	0.347	0.777	0.347	0.974	0.480
-	-	-	-	-	-	-	-	-	-	-
-	-	-	-	-	-	-	-	-	-	-
1,361	1,192									

Logistic Regression considering shear types 0,1,2,3				
	Power Law	Flat	LLJ	Others
Power Law	1,229	15	35	0
Flat	71	13	0	0
LLJ	92	0	136	0
Others	17	0	2	0

Total # of subjects:	1,610
Correct classified:	1,378
Wrong classified:	232
Accuracy:	85.590%
Error:	14.410%
Cohen's kappa (k):	0.501

Agreement	By Chance	TruePositives	Type II error FalsePositives	Type I error FalseNegatives	TrueNegatives	Recall	Precision	Sensitivity	Specifity	F-measure
1,229	1,119	1,229	180	50	151	0.961	0.872	0.961	0.456	0.914
13	1	13	15	71	1,511	0.155	0.464	0.155	0.990	0.232
136	24	136	37	92	1,345	0.596	0.786	0.596	0.973	0.678
-	-	-	-	19	1,357	-	-	-	1.000	-
1,378	1,145									

Tree Ensemble

Tree Ensemble				
	Power Law	No Power Law		
Power Law	1,252	27		
No Power Law	101	230		
Area Under Curve:		**0.928**		

Total # of subjects:	**1,610**
Correct classified:	**1,482**
Wrong classified:	**128**
Accuracy:	**92.050%**
Error:	**7.950%**
Cohen's kappa (k):	**0.735**

					Type II error	Type I error					
Agreement	**By Chance**	**TruePositives**	**FalsePositives**	**FalseNegatives**	**TrueNegatives**	**Recall**	**Precision**	**Sensitivity**	**Specifity**	**F-measure**
1,252	1,075	1,252	101	27	230	0.979	0.925	0.979	0.695	0.951
230	53	230	27	101	1,252	0.695	0.895	0.695	0.979	0.782
-	-	-	-	-	-	-	-	-	-	-
-	-	-	-	-	-	-	-	-	-	-
1,482	**1,128**									

Tree Ensemble considering shear types 0,1,2,3				
	Power Law	Flat	LLJ	Others
Power Law	1,256	6	16	1
Flat	49	35	0	0
LLJ	39	0	188	1
Others	11	1	2	5

Total # of subjects:	**1,610**
Correct classified:	**1,484**
Wrong classified:	**126**
Accuracy:	**92.174%**
Error:	**7.826%**
Cohen's kappa (k):	**0.749**

					Type II error	Type I error					
Agreement	**By Chance**	**TruePositives**	**FalsePositives**	**FalseNegatives**	**TrueNegatives**	**Recall**	**Precision**	**Sensitivity**	**Specifity**	**F-measure**
1,256	1,076	1,256	99	23	232	0.982	0.927	0.982	0.701	0.954
35	2	35	7	49	1,519	0.417	0.833	0.417	0.995	0.556
188	29	188	18	40	1,364	0.825	0.913	0.825	0.987	0.866
5	0	5	2	14	1,466	0.263	0.714	0.263	0.999	0.385
1,484	**1,108**									

K Nearest Neighbor models.

K Nearest Neighbor				
	Power Law	No Power Law		
Power Law	1,234	45		
No Power Law	110	221		
Area Under Curve:		**0.906**		

Total # of subjects:	**1,610**
Correct classified:	**1,455**
Wrong classified:	**155**
Accuracy:	**90.373%**
Error:	**9.627%**
Cohen's kappa (k):	**0.682**

					Type II error	Type I error					
Agreement	**By Chance**	**TruePositives**	**FalsePositives**	**FalseNegatives**	**TrueNegatives**	**Recall**	**Precision**	**Sensitivity**	**Specifity**	**F-measure**
1,234	1,068	1,234	110	45	221	0.965	0.918	0.965	0.668	0.941
221	55	221	45	110	1,234	0.668	0.831	0.668	0.965	0.740
-	-	-	-	-	-	-	-	-	-	-
-	-	-	-	-	-	-	-	-	-	-
1,455	**1,122**									

K Nearest Neighbor considering shear types 0,1,2,3				
	Power Law	Flat	LLJ	Others
Power Law	1,237	9	28	5
Flat	51	31	2	0
LLJ	51	0	176	1
Others	11	1	1	6

Total # of subjects:	**1,610**
Correct classified:	**1,450**
Wrong classified:	**160**
Accuracy:	**90.062%**
Error:	**9.938%**
Cohen's kappa (k):	**0.684**

					Type II error	Type I error					
Agreement	**By Chance**	**TruePositives**	**FalsePositives**	**FalseNegatives**	**TrueNegatives**	**Recall**	**Precision**	**Sensitivity**	**Specifity**	**F-measure**
1,237	1,072	1,237	113	42	218	0.967	0.916	0.967	0.659	0.941
31	2	31	10	53	1,516	0.369	0.756	0.369	0.993	0.496
176	29	176	31	52	1,351	0.772	0.850	0.772	0.978	0.809
6	0	6	6	13	1,430	0.316	0.500	0.316	0.996	0.387
1,450	**1,104**									

End of “Results of testing the models without removing correlated columns.” subsection.

Results of testing the models removing correlated columns.

Random Forest

Random Forest				
	Power Law	No Power Law		
Power Law	1,243	36		
No Power Law	108	223		
Area Under Curve:		**0.924**		

Total # of subjects:	**1,610**
Correct classified:	**1,466**
Wrong classified:	**144**
Accuracy:	**91.056%**
Error:	**8.944%**
Cohen's kappa (k):	**0.702**

					Type II error	**Type I error**					
Agreement	**By Chance**	**TruePositives**	**FalsePositives**	**FalseNegatives**	**TrueNegatives**	**Recall**	**Precision**	**Sensitivity**	**Specifity**	**F-measure**
1,243	1,073	1,243	108	36	223	0.972	0.920	0.972	0.674	0.945
223	53	223	36	108	1,243	0.674	0.861	0.674	0.972	0.756
-	-	-	-	-	-	-	-	-	-	-
-	-	-	-	-	-	-	-	-	-	-
1,466	**1,126**									

Random Forest considering shear types 0,1,2,3				
	Power Law	Flat	LLJ	Others
Power Law	1,256	6	15	2
Flat	50	34	0	0
LLJ	43	0	185	0
Others	14	1	2	2

Total # of subjects:	**1,610**
Correct classified:	**1,477**
Wrong classified:	**133**
Accuracy:	**91.739%**
Error:	**8.261%**
Cohen's kappa (k):	**0.732**

					Type II error	**Type I error**					
Agreement	**By Chance**	**TruePositives**	**FalsePositives**	**FalseNegatives**	**TrueNegatives**	**Recall**	**Precision**	**Sensitivity**	**Specifity**	**F-measure**
1,256	1,083	1,256	107	23	224	0.982	0.921	0.982	0.677	0.951
34	2	34	7	50	1,519	0.405	0.829	0.405	0.995	0.544
185	29	185	17	43	1,365	0.811	0.916	0.811	0.988	0.860
2	0	2	2	17	1,456	0.105	0.500	0.105	0.999	0.174
1,477	**1,114**									

Decision Tree (using R)

Decision Tree - R				
	Power Law	No Power Law		
Power Law	1,232	47		
No Power Law	174	157		
Area Under Curve:		**0.753**		

Total # of subjects:	**1,610**
Correct classified:	**1,389**
Wrong classified:	**221**
Accuracy:	**86.273%**
Error:	**13.727%**
Cohen's kappa (k):	**0.510**

					Type II error	**Type I error**					
Agreement	**By Chance**	**TruePositives**	**FalsePositives**	**FalseNegatives**	**TrueNegatives**	**Recall**	**Precision**	**Sensitivity**	**Specifity**	**F-measure**
1,232	1,117	1,232	174	47	157	0.963	0.876	0.963	0.474	0.918
157	42	157	47	174	1,232	0.474	0.770	0.474	0.963	0.587
-	-	-	-	-	-	-	-	-	-	-
-	-	-	-	-	-	-	-	-	-	-
1,389	**1,159**									

Decision Tree - R considering shear types 0,1,2,3				
	Power Law	Flat	LLJ	Others
Power Law	1,165	46	54	14
Flat	41	42	1	0
LLJ	49	1	175	3
Others	16	1	1	1

Total # of subjects:	**1,610**
Correct classified:	**1,383**
Wrong classified:	**227**
Accuracy:	**85.901%**
Error:	**14.099%**
Cohen's kappa (k):	**0.597**

					Type II error	**Type I error**					
Agreement	**By Chance**	**TruePositives**	**FalsePositives**	**FalseNegatives**	**TrueNegatives**	**Recall**	**Precision**	**Sensitivity**	**Specifity**	**F-measure**
1,165	1,010	1,165	106	114	225	0.911	0.917	0.911	0.680	0.914
42	5	42	48	42	1,478	0.500	0.467	0.500	0.969	0.483
175	33	175	56	53	1,326	0.768	0.758	0.768	0.959	0.763
1	0	1	17	18	1,347	0.053	0.056	0.053	0.988	0.054
1,383	**1,047**									

Naive Bayes

Naïve Bayes					Total # of subjects:	1,610				Type II error	Type I error						
	Power Law	No Power Law			Correct classified:	1,188	Agreement	By Chance	TruePositives	FalsePositives	FalseNegatives	TrueNegatives	Recall	Precision	Sensitivity	Specifity	F-measure
Power Law	952	327			Wrong classified:	422	952	832	952	95	327	236	0.744	0.909	0.744	0.713	0.819
No Power Law	95	236			Accuracy:	73.789%	236	116	236	327	95	952	0.713	0.419	0.713	0.744	0.528
					Error:	26.211%	-	-	-	-	-	-	-	-	-	-	-
					Cohen's kappa (k):	0.363	-	-	-	-	-	-	-	-	-	-	-
Area Under Curve:		0.791					1,188	947									

Naïve Bayes considering shear types 0,1,2,3					Total # of subjects:	1,610				Type II error	Type I error						
	Power Law	Flat	LLJ	Others	Correct classified:	1,251	Agreement	By Chance	TruePositives	FalsePositives	FalseNegatives	TrueNegatives	Recall	Precision	Sensitivity	Specifity	F-measure
Power Law	1,059	5	205	10	Wrong classified:	359	1,059	931	1,059	113	220	218	0.828	0.904	0.828	0.659	0.864
Flat	64	6	14	0	Accuracy:	77.702%	6	1	6	5	78	1,521	0.071	0.545	0.071	0.997	0.126
LLJ	38	0	183	7	Error:	22.298%	183	58	183	224	45	1,158	0.803	0.450	0.803	0.838	0.576
Others	11	0	5	3	Cohen's kappa (k):	0.421	3	0	3	17	16	1,213	0.158	0.150	0.158	0.986	0.154
							1,251	989									

Logistic Regression

Logistic Regression					Total # of subjects:	1,610				Type II error	Type I error						
	Power Law	No Power Law			Correct classified:	1,362	Agreement	By Chance	TruePositives	FalsePositives	FalseNegatives	TrueNegatives	Recall	Precision	Sensitivity	Specifity	F-measure
Power Law	1,252	27			Wrong classified:	248	1,252	1,170	1,252	221	27	110	0.979	0.850	0.979	0.332	0.910
No Power Law	221	110			Accuracy:	84.596%	110	28	110	27	221	1,252	0.332	0.803	0.332	0.979	0.470
					Error:	15.404%	-	-	-	-	-	-	-	-	-	-	-
					Cohen's kappa (k):	0.398	-	-	-	-	-	-	-	-	-	-	-
Area Under Curve:		0.777					1,362	1,198									

Logistic Regression considering shear types 0,1,2,3					Total # of subjects:	1,610				Type II error	Type I error						
	Power Law	Flat	LLJ	Others	Correct classified:	1,377	Agreement	By Chance	TruePositives	FalsePositives	FalseNegatives	TrueNegatives	Recall	Precision	Sensitivity	Specifity	F-measure
Power Law	1,233	14	32	0	Wrong classified:	233	1,233	1,126	1,233	184	46	147	0.964	0.870	0.964	0.444	0.915
Flat	74	9	1	0	Accuracy:	85.528%	9	1	9	14	75	1,512	0.107	0.391	0.107	0.991	0.168
LLJ	93	0	135	0	Error:	14.472%	135	24	135	35	93	1,347	0.592	0.794	0.592	0.975	0.678
Others	17	0	2	0	Cohen's kappa (k):	0.492	-	-	-	-	19	1,356	-	-	-	1.000	-
							1,377	1,151									

Tree Ensemble

Tree Ensemble				
	Power Law	No Power Law		
Power Law	1,246	33		
No Power Law	105	226		
Area Under Curve:		0.931		

Total # of subjects:	1,610
Correct classified:	1,472
Wrong classified:	138
Accuracy:	91.429%
Error:	8.571%
Cohen's kappa (k):	0.715

Agreement	By Chance	TruePositives	Type II error FalsePositives	Type I error FalseNegatives	TrueNegatives	Recall	Precision	Sensitivity	Specifity	F-measure
1,246	1,073	1,246	105	33	226	0.974	0.922	0.974	0.683	0.948
226	53	226	33	105	1,246	0.683	0.873	0.683	0.974	0.766
-	-	-	-	-	-	-	-	-	-	-
-	-	-	-	-	-	-	-	-	-	-
1,472	1,126									

Tree Ensemble considering shear types 0,1,2,3				
	Power Law	Flat	LLJ	Others
Power Law	1,252	7	17	3
Flat	52	32	0	0
LLJ	44	0	184	0
Others	14	1	1	3

Total # of subjects:	1,610
Correct classified:	1,471
Wrong classified:	139
Accuracy:	91.366%
Error:	8.634%
Cohen's kappa (k):	0.720

Agreement	By Chance	TruePositives	Type II error FalsePositives	Type I error FalseNegatives	TrueNegatives	Recall	Precision	Sensitivity	Specifity	F-measure
1,252	1,082	1,252	110	27	221	0.979	0.919	0.979	0.668	0.948
32	2	32	8	52	1,518	0.381	0.800	0.381	0.995	0.516
184	29	184	18	44	1,364	0.807	0.911	0.807	0.987	0.856
3	0	3	3	16	1,451	0.158	0.500	0.158	0.998	0.240
1,471	1,113									

K Nearest Neighbor models.

K Nearest Neighbor				
	Power Law	No Power Law		
Power Law	1,232	47		
No Power Law	126	205		
Area Under Curve:		0.877		

Total # of subjects:	1,610
Correct classified:	1,437
Wrong classified:	173
Accuracy:	89.255%
Error:	10.745%
Cohen's kappa (k):	0.639

Agreement	By Chance	TruePositives	Type II error FalsePositives	Type I error FalseNegatives	TrueNegatives	Recall	Precision	Sensitivity	Specifity	F-measure
1,232	1,079	1,232	126	47	205	0.963	0.907	0.963	0.619	0.934
205	52	205	47	126	1,232	0.619	0.813	0.619	0.963	0.703
-	-	-	-	-	-	-	-	-	-	-
-	-	-	-	-	-	-	-	-	-	-
1,437	1,131									

K Nearest Neighbor considering shear types 0,1,2,3				
	Power Law	Flat	LLJ	Others
Power Law	1,238	9	31	1
Flat	62	21	1	0
LLJ	57	0	168	3
Others	12	1	2	4

Total # of subjects:	1,610
Correct classified:	1,431
Wrong classified:	179
Accuracy:	88.882%
Error:	11.118%
Cohen's kappa (k):	0.636

Agreement	By Chance	TruePositives	Type II error FalsePositives	Type I error FalseNegatives	TrueNegatives	Recall	Precision	Sensitivity	Specifity	F-measure
1,238	1,088	1,238	131	41	200	0.968	0.904	0.968	0.604	0.935
21	2	21	10	63	1,516	0.250	0.677	0.250	0.993	0.365
168	29	168	34	60	1,348	0.737	0.832	0.737	0.975	0.781
4	0	4	4	15	1,410	0.211	0.500	0.211	0.997	0.296
1,431	1,118									

End of “Results of testing the models removing correlated columns.” subsection.

Results of testing the models removing correlated columns, and removing 1st and last 3 days of data.

Random Forest

Random Forest				
	Power Law	No Power Law		
Power Law	1,102	0		
No Power Law	49	1		
Area Under Curve:		**0.757**		

Total # of subjects:	**1,152**
Correct classified:	**1,103**
Wrong classified:	**49**
Accuracy:	**95.747%**
Error:	**4.253%**
Cohen's kappa (k):	**0.038**

					Type II error	Type I error					
Agreement	By Chance	TruePositives	FalsePositives	FalseNegatives	TrueNegatives	Recall	Precision	Sensitivity	Specifity	F-measure	
1,102	1,101	1,102	49	-	1	1.000	0.957	1.000	0.020	0.978	
1	0	1	-	49	1,102	0.020	1.000	0.020	1.000	0.039	
-	-	-	-	-	-	-	-	-	-	-	
-	-	-	-	-	-	-	-	-	-	-	
1,103	**1,101**										

Random Forest considering shear types 0,1,2,3				
	Power Law	Flat	LLJ	Others
Power Law	1,102	0	0	0
Flat	26	0	0	0
LLJ	21	0	0	0
Others	3	0	0	0

Total # of subjects:	**1,152**
Correct classified:	**1,102**
Wrong classified:	**50**
Accuracy:	**95.660%**
Error:	**4.340%**
Cohen's kappa (k):	**0.000**

					Type II error	Type I error					
Agreement	By Chance	TruePositives	FalsePositives	FalseNegatives	TrueNegatives	Recall	Precision	Sensitivity	Specifity	F-measure	
1,102	1,102	1,102	50	-	-	1.000	0.957	1.000	-	0.978	
-	-	-	-	26	1,126	-	-	-	1.000	-	
-	-	-	-	21	1,131	-	-	-	1.000	-	
-	-	-	-	3	1,099	-	-	-	1.000	-	
1,102	**1,102**										

Decision Tree (using R)

Decision Tree - R				
	Power Law	No Power Law		
Power Law	1,102	0		
No Power Law	50	0		
Area Under Curve:		**0.500**		

Total # of subjects:	**1,152**
Correct classified:	**1,102**
Wrong classified:	**50**
Accuracy:	**95.660%**
Error:	**4.340%**
Cohen's kappa (k):	**0.000**

					Type II error	Type I error					
Agreement	By Chance	TruePositives	FalsePositives	FalseNegatives	TrueNegatives	Recall	Precision	Sensitivity	Specifity	F-measure	
1,102	1,102	1,102	50	-	-	1.000	0.957	1.000	-	0.978	
-	-	-	-	50	1,102	-	-	-	1.000	-	
-	-	-	-	-	-	-	-	-	-	-	
-	-	-	-	-	-	-	-	-	-	-	
1,102	**1,102**										

Decision Tree - R considering shear types 0,1,2,3				
	Power Law	Flat	LLJ	Others
Power Law	1,071	18	12	1
Flat	22	3	1	0
LLJ	20	0	1	0
Others	3	0	0	0

Total # of subjects:	**1,152**
Correct classified:	**1,075**
Wrong classified:	**77**
Accuracy:	**93.316%**
Error:	**6.684%**
Cohen's kappa (k):	**0.080**

					Type II error	Type I error					
Agreement	By Chance	TruePositives	FalsePositives	FalseNegatives	TrueNegatives	Recall	Precision	Sensitivity	Specifity	F-measure	
1,071	1,068	1,071	45	31	5	0.972	0.960	0.972	0.100	0.966	
3	0	3	18	23	1,108	0.115	0.143	0.115	0.984	0.128	
1	0	1	13	20	1,118	0.048	0.071	0.048	0.989	0.057	
-	0	-	1	3	1,071	-	-	-	0.999	-	
1,075	**1,068**										

Naive Bayes

Naïve Bayes					Total # of subjects:	1,152				Type II error	Type I error						
	Power Law	No Power Law			Correct classified:	1,100	Agreement	By Chance	TruePositives	FalsePositives	FalseNegatives	TrueNegatives	Recall	Precision	Sensitivity	Specifity	F-measure
Power Law	1,100	2			Wrong classified:	52	1,100	1,100	1,100	50	2	-	0.998	0.957	0.998	-	0.977
No Power Law	50	0			Accuracy:	95.486%	-	0	-	2	50	1,100	-	-	-	0.998	-
					Error:	4.514%	-	-	-	-	-	-	-	-	-	-	-
					Cohen's kappa (k):	-0.003	-	-	-	-	-	-	-	-	-	-	-
Area Under Curve:		0.577					1,100	1,100									

Naïve Bayes considering shear types 0,1,2,3					Total # of subjects:	1,152				Type II error	Type I error						
	Power Law	Flat	LLJ	Others	Correct classified:	1,088	Agreement	By Chance	TruePositives	FalsePositives	FalseNegatives	TrueNegatives	Recall	Precision	Sensitivity	Specifity	F-measure
Power Law	1,087	11	4	0	Wrong classified:	64	1,087	1,087	1,087	49	15	1	0.986	0.957	0.986	0.020	0.971
Flat	20	1	0	0	Accuracy:	94.444%	1	0	1	11	20	1,120	0.048	0.083	0.048	0.990	0.061
LLJ	3	0	0	0	Error:	5.556%	-	0	-	4	3	1,145	-	-	-	0.997	-
Others	26	0	0	0	Cohen's kappa (k):	0.017	-	-	-	-	26	1,062	-	-	-	1.000	-
							1,088	1,087									

Logistic Regression

Logistic Regression					Total # of subjects:	1,152				Type II error	Type I error						
	Power Law	No Power Law			Correct classified:	1,102	Agreement	By Chance	TruePositives	FalsePositives	FalseNegatives	TrueNegatives	Recall	Precision	Sensitivity	Specifity	F-measure
Power Law	1,102	0			Wrong classified:	50	1,102	1,102	1,102	50	-	-	1.000	0.957	1.000	-	0.978
No Power Law	50	0			Accuracy:	95.660%	-	-	-	-	50	1,102	-	-	-	1.000	-
					Error:	4.340%	-	-	-	-	-	-	-	-	-	-	-
					Cohen's kappa (k):	0.000	-	-	-	-	-	-	-	-	-	-	-
Area Under Curve:		0.619					1,102	1,102									

Logistic Regression considering shear types 0,1,2,3					Total # of subjects:	1,152				Type II error	Type I error						
	Power Law	Flat	LLJ	Others	Correct classified:	1,102	Agreement	By Chance	TruePositives	FalsePositives	FalseNegatives	TrueNegatives	Recall	Precision	Sensitivity	Specifity	F-measure
Power Law	1,102	0	0	0	Wrong classified:	50	1,102	1,102	1,102	50	-	-	1.000	0.957	1.000	-	0.978
Flat	26	0	0	0	Accuracy:	95.660%	-	-	-	-	26	1,126	-	-	-	1.000	-
LLJ	21	0	0	0	Error:	4.340%	-	-	-	-	21	1,131	-	-	-	1.000	-
Others	3	0	0	0	Cohen's kappa (k):	0.000	-	-	-	-	3	1,099	-	-	-	1.000	-
							1,102	1,102									

Tree Ensemble

Tree Ensemble				
	Power Law	No Power Law		
Power Law	1,102	0		
No Power Law	48	2		
Area Under Curve:		0.777		

Total # of subjects:	1,152
Correct classified:	1,104
Wrong classified:	48
Accuracy:	95.833%
Error:	4.167%
Cohen's kappa (k):	0.074

				Type II error	Type I error						
Agreement	By Chance	TruePositives	FalsePositives	FalseNegatives	TrueNegatives	Recall	Precision	Sensitivity	Specifity	F-measure
1,102	1,100	1,102	48	-	2	1.000	0.958	1.000	0.040	0.979
2	0	2	-	48	1,102	0.040	1.000	0.040	1.000	0.077
-	-	-	-	-	-	-	-	-	-	-
-	-	-	-	-	-	-	-	-	-	-
1,104	1,100									

Tree Ensemble considering shear types 0,1,2,3				
	Power Law	Flat	LLJ	Others
Power Law	1,102	0	0	0
Flat	20	1	0	0
LLJ	26	0	0	0
Others	3	0	0	0

Total # of subjects:	1,152
Correct classified:	1,103
Wrong classified:	49
Accuracy:	95.747%
Error:	4.253%
Cohen's kappa (k):	0.038

				Type II error	Type I error						
Agreement	By Chance	TruePositives	FalsePositives	FalseNegatives	TrueNegatives	Recall	Precision	Sensitivity	Specifity	F-measure
1,102	1,101	1,102	49	-	1	1.000	0.957	1.000	0.020	0.978
1	0	1	-	20	1,131	0.048	1.000	0.048	1.000	0.091
-	-	-	-	26	1,126	-	-	-	1.000	-
-	-	-	-	3	1,100	-	-	-	1.000	-
1,103	1,101									

K Nearest Neighbor models.

K Nearest Neighbor				
	Power Law	No Power Law		
Power Law	1,102	0		
No Power Law	49	1		
Area Under Curve:		0.713		

Total # of subjects:	1,152
Correct classified:	1,103
Wrong classified:	49
Accuracy:	95.747%
Error:	4.253%
Cohen's kappa (k):	0.038

				Type II error	Type I error						
Agreement	By Chance	TruePositives	FalsePositives	FalseNegatives	TrueNegatives	Recall	Precision	Sensitivity	Specifity	F-measure
1,102	1,101	1,102	49	-	1	1.000	0.957	1.000	0.020	0.978
1	0	1	-	49	1,102	0.020	1.000	0.020	1.000	0.039
-	-	-	-	-	-	-	-	-	-	-
-	-	-	-	-	-	-	-	-	-	-
1,103	1,101									

K Nearest Neighbor considering shear types 0,1,2,3				
	Power Law	Flat	LLJ	Others
Power Law	1,102	0	0	0
Flat	26	0	0	0
LLJ	20	0	1	0
Others	3	0	0	0

Total # of subjects:	1,152
Correct classified:	1,103
Wrong classified:	49
Accuracy:	95.747%
Error:	4.253%
Cohen's kappa (k):	0.038

				Type II error	Type I error						
Agreement	By Chance	TruePositives	FalsePositives	FalseNegatives	TrueNegatives	Recall	Precision	Sensitivity	Specifity	F-measure
1,102	1,101	1,102	49	-	1	1.000	0.957	1.000	0.020	0.978
-	-	-	-	26	1,126	-	-	-	1.000	-
1	0	1	-	20	1,131	0.048	1.000	0.048	1.000	0.091
-	-	-	-	3	1,100	-	-	-	1.000	-
1,103	1,101									

End of “Results of testing the models removing correlated columns, and removing 1st and last 3 days of data.” subsection.

Results of testing the models removing correlated columns and LLJ data rows.

Random Forest

Random Forest				
	Power Law	No Power Law		
Power Law	1,264	11		
No Power Law	69	30		
Area Under Curve:		**0.900**		

Total # of subjects:	**1,374**
Correct classified:	**1,294**
Wrong classified:	**80**
Accuracy:	**94.178%**
Error:	**5.822%**
Cohen's kappa (k):	**0.403**

Agreement	By Chance	TruePositives	Type II error FalsePositives	Type I error FalseNegatives	TrueNegatives	Recall	Precision	Sensitivity	Specifity	F-measure
1,264	1,237	1,264	69	11	30	0.991	0.948	0.991	0.303	0.969
30	3	30	11	69	1,264	0.303	0.732	0.303	0.991	0.429
-	-	-	-	-	-	-	-	-	-	-
-	-	-	-	-	-	-	-	-	-	-
1,294	**1,240**									

Random Forest considering shear types 0,1,2,3				
	Power Law	Flat	LLJ	Others
Power Law	1,268	6	1	0
Flat	49	21	0	0
LLJ	22	0	7	0
Others	0	0	0	0

Total # of subjects:	**1,374**
Correct classified:	**1,296**
Wrong classified:	**78**
Accuracy:	**94.323%**
Error:	**5.677%**
Cohen's kappa (k):	**0.400**

Agreement	By Chance	TruePositives	Type II error FalsePositives	Type I error FalseNegatives	TrueNegatives	Recall	Precision	Sensitivity	Specifity	F-measure
1,268	1,243	1,268	71	7	28	0.995	0.947	0.995	0.283	0.970
21	1	21	6	49	1,298	0.300	0.778	0.300	0.995	0.433
7	0	7	1	22	1,344	0.241	0.875	0.241	0.999	0.378
-	-	-	-	-	-	-	-	-	-	-
1,296	**1,244**									

Decision Tree (using R)

Decision Tree - R				
	Power Law	No Power Law		
Power Law	1,267	8		
No Power Law	87	12		
Area Under Curve:		**0.655**		

Total # of subjects:	**1,374**
Correct classified:	**1,279**
Wrong classified:	**95**
Accuracy:	**93.086%**
Error:	**6.914%**
Cohen's kappa (k):	**0.182**

Agreement	By Chance	TruePositives	Type II error FalsePositives	Type I error FalseNegatives	TrueNegatives	Recall	Precision	Sensitivity	Specifity	F-measure
1,267	1,256	1,267	87	8	12	0.994	0.936	0.994	0.121	0.964
12	1	12	8	87	1,267	0.121	0.600	0.121	0.994	0.202
-	-	-	-	-	-	-	-	-	-	-
-	-	-	-	-	-	-	-	-	-	-
1,279	**1,258**									

Decision Tree - R considering shear types 0,1,2,3				
	Power Law	Flat	LLJ	Others
Power Law	1,208	51	16	0
Flat	37	32	1	0
LLJ	22	1	6	0
Others	0	0	0	0

Total # of subjects:	**1,374**
Correct classified:	**1,246**
Wrong classified:	**128**
Accuracy:	**90.684%**
Error:	**9.316%**
Cohen's kappa (k):	**0.339**

Agreement	By Chance	TruePositives	Type II error FalsePositives	Type I error FalseNegatives	TrueNegatives	Recall	Precision	Sensitivity	Specifity	F-measure
1,208	1,176	1,208	59	67	40	0.947	0.953	0.947	0.404	0.950
32	4	32	52	38	1,252	0.457	0.381	0.457	0.960	0.416
6	0	6	17	23	1,328	0.207	0.261	0.207	0.987	0.231
-	-	-	-	-	-	-	-	-	-	-
1,246	**1,180**									

Naive Bayes

Naïve Bayes					Total # of subjects:	1,374				Type II error	Type I error						
	Power Law	No Power Law			Correct classified:	1,273	Agreement	By Chance	TruePositives	FalsePositives	FalseNegatives	TrueNegatives	Recall	Precision	Sensitivity	Specifity	F-measure
Power Law	1,257	18			Wrong classified:	101	1,257	1,243	1,257	83	18	16	0.986	0.938	0.986	0.162	0.961
No Power Law	83	16			Accuracy:	92.649%	16	2	16	18	83	1,257	0.162	0.471	0.162	0.986	0.241
					Error:	7.351%	-	-	-	-	-	-	-	-	-	-	-
					Cohen's kappa (k):	0.212	-	-	-	-	-	-	-	-	-	-	-
Area Under Curve:		0.732					1,273	1,246									

Naïve Bayes considering shear types 0,1,2,3					Total # of subjects:	1,374				Type II error	Type I error						
	Power Law	Flat	LLJ	Others	Correct classified:	1,255	Agreement	By Chance	TruePositives	FalsePositives	FalseNegatives	TrueNegatives	Recall	Precision	Sensitivity	Specifity	F-measure
Power Law	1,241	7	27	0	Wrong classified:	119	1,241	1,230	1,241	84	34	15	0.973	0.937	0.973	0.152	0.955
Flat	59	10	1	0	Accuracy:	91.339%	10	1	10	7	60	1,297	0.143	0.588	0.143	0.995	0.230
LLJ	25	0	4	0	Error:	8.661%	4	1	4	28	25	1,317	0.138	0.125	0.138	0.979	0.131
Others	0	0	0	0	Cohen's kappa (k):	0.167	-	-	-	-	-	-	-	-	-	-	-
							1,255	1,231									

Logistic Regression

Logistic Regression					Total # of subjects:	1,374				Type II error	Type I error						
	Power Law	No Power Law			Correct classified:	1,276	Agreement	By Chance	TruePositives	FalsePositives	FalseNegatives	TrueNegatives	Recall	Precision	Sensitivity	Specifity	F-measure
Power Law	1,272	3			Wrong classified:	98	1,272	1,269	1,272	95	3	4	0.998	0.931	0.998	0.040	0.963
No Power Law	95	4			Accuracy:	92.868%	4	1	4	3	95	1,272	0.040	0.571	0.040	0.998	0.075
					Error:	7.132%	-	-	-	-	-	-	-	-	-	-	-
					Cohen's kappa (k):	0.067	-	-	-	-	-	-	-	-	-	-	-
Area Under Curve:		0.715					1,276	1,269									

Logistic Regression considering shear types 0,1,2,3					Total # of subjects:	1,374				Type II error	Type I error						
	Power Law	Flat	LLJ	Others	Correct classified:	1,273	Agreement	By Chance	TruePositives	FalsePositives	FalseNegatives	TrueNegatives	Recall	Precision	Sensitivity	Specifity	F-measure
Power Law	1,264	11	0	0	Wrong classified:	101	1,264	1,256	1,264	90	11	9	0.991	0.934	0.991	0.091	0.962
Flat	61	9	0	0	Accuracy:	92.649%	9	1	9	11	61	1,293	0.129	0.450	0.129	0.992	0.200
LLJ	0	0	0	0	Error:	7.351%	-	-	-	-	-	-	-	-	-	-	-
Others	29	0	0	0	Cohen's kappa (k):	0.133	-	-	-	-	29	1,244	-	-	-	1.000	-
							1,273	1,257									

Tree Ensemble

Tree Ensemble				
	Power Law	No Power Law		
Power Law	1,268	7		
No Power Law	70	29		
Area Under Curve:		**0.878**		

Total # of subjects:	**1,374**
Correct classified:	**1,297**
Wrong classified:	**77**
Accuracy:	**94.396%**
Error:	**5.604%**
Cohen's kappa (k):	**0.407**

					Type II error	**Type I error**					
Agreement	**By Chance**	**TruePositives**	**FalsePositives**	**FalseNegatives**	**TrueNegatives**	**Recall**	**Precision**	**Sensitivity**	**Specifity**	**F-measure**
1,268	1,242	1,268	70	7	29	0.995	0.948	0.995	0.293	0.971
29	3	29	7	70	1,268	0.293	0.806	0.293	0.995	0.430
-	-	-	-	-	-	-	-	-	-	-
-	-	-	-	-	-	-	-	-	-	-
1,297	**1,244**									

Tree Ensemble considering shear types 0,1,2,3				
	Power Law	Flat	LLJ	Others
Power Law	1,269	6	0	0
Flat	50	19	1	0
LLJ	22	0	7	0
Others	0	0	0	0

Total # of subjects:	**1,374**
Correct classified:	**1,295**
Wrong classified:	**79**
Accuracy:	**94.250%**
Error:	**5.750%**
Cohen's kappa (k):	**0.384**

					Type II error	**Type I error**					
Agreement	**By Chance**	**TruePositives**	**FalsePositives**	**FalseNegatives**	**TrueNegatives**	**Recall**	**Precision**	**Sensitivity**	**Specifity**	**F-measure**
1,269	1,244	1,269	72	6	27	0.995	0.946	0.995	0.273	0.970
19	1	19	6	51	1,298	0.271	0.760	0.271	0.995	0.400
7	0	7	1	22	1,344	0.241	0.875	0.241	0.999	0.378
-	-	-	-	-	-	-	-	-	-	-
1,295	**1,246**									

K Nearest Neighbor models.

K Nearest Neighbor				
	Power Law	No Power Law		
Power Law	1,266	9		
No Power Law	76	23		
Area Under Curve:		**0.842**		

Total # of subjects:	**1,374**
Correct classified:	**1,289**
Wrong classified:	**85**
Accuracy:	**93.814%**
Error:	**6.186%**
Cohen's kappa (k):	**0.327**

					Type II error	**Type I error**					
Agreement	**By Chance**	**TruePositives**	**FalsePositives**	**FalseNegatives**	**TrueNegatives**	**Recall**	**Precision**	**Sensitivity**	**Specifity**	**F-measure**
1,266	1,245	1,266	76	9	23	0.993	0.943	0.993	0.232	0.968
23	2	23	9	76	1,266	0.232	0.719	0.232	0.993	0.351
-	-	-	-	-	-	-	-	-	-	-
-	-	-	-	-	-	-	-	-	-	-
1,289	**1,248**									

K Nearest Neighbor considering shear types 0,1,2,3				
	Power Law	Flat	LLJ	Others
Power Law	1,266	9	0	0
Flat	49	20	0	1
LLJ	0	0	0	0
Others	27	0	0	2

Total # of subjects:	**1,374**
Correct classified:	**1,288**
Wrong classified:	**86**
Accuracy:	**93.741%**
Error:	**6.259%**
Cohen's kappa (k):	**0.324**

					Type II error	**Type I error**					
Agreement	**By Chance**	**TruePositives**	**FalsePositives**	**FalseNegatives**	**TrueNegatives**	**Recall**	**Precision**	**Sensitivity**	**Specifity**	**F-measure**
1,266	1,245	1,266	76	9	23	0.993	0.943	0.993	0.232	0.968
20	1	20	9	50	1,295	0.286	0.690	0.286	0.993	0.404
-	-	-	-	-	-	-	-	-	-	-
2	0	2	1	27	1,260	0.069	0.667	0.069	0.999	0.125
1,288	**1,247**									

End of “Results of testing the models removing correlated columns and LLJ data rows.” subsection.

Results of testing the models removing correlated columns and normalizing columns before applying models.

Random Forest

Random Forest				
	Power Law	No Power Law		
Power Law	1,243	36		
No Power Law	108	223		
Area Under Curve:		**0.924**		

Total # of subjects:	**1,610**
Correct classified:	**1,466**
Wrong classified:	**144**
Accuracy:	**91.056%**
Error:	**8.944%**
Cohen's kappa (k):	**0.702**

Agreement	By Chance	TruePositives	Type II error FalsePositives	Type I error FalseNegatives	TrueNegatives	Recall	Precision	Sensitivity	Specifity	F-measure
1,243	1,073	1,243	108	36	223	0.972	0.920	0.972	0.674	0.945
223	53	223	36	108	1,243	0.674	0.861	0.674	0.972	0.756
-	-	-	-	-	-	-	-	-	-	-
-	-	-	-	-	-	-	-	-	-	-
1,466	**1,126**									

Random Forest considering shear types 0,1,2,3				
	Power Law	Flat	LLJ	Others
Power Law	1,256	6	15	2
Flat	50	34	0	0
LLJ	43	0	185	0
Others	14	1	2	2

Total # of subjects:	**1,610**
Correct classified:	**1,477**
Wrong classified:	**133**
Accuracy:	**91.739%**
Error:	**8.261%**
Cohen's kappa (k):	**0.732**

Agreement	By Chance	TruePositives	Type II error FalsePositives	Type I error FalseNegatives	TrueNegatives	Recall	Precision	Sensitivity	Specifity	F-measure
1,256	1,083	1,256	107	23	224	0.982	0.921	0.982	0.677	0.951
34	2	34	7	50	1,519	0.405	0.829	0.405	0.995	0.544
185	29	185	17	43	1,365	0.811	0.916	0.811	0.988	0.860
2	0	2	2	17	1,456	0.105	0.500	0.105	0.999	0.174
1,477	**1,114**									

Decision Tree (using R)

Decision Tree - R				
	Power Law	No Power Law		
Power Law	1,232	47		
No Power Law	174	157		
Area Under Curve:		**0.753**		

Total # of subjects:	**1,610**
Correct classified:	**1,389**
Wrong classified:	**221**
Accuracy:	**86.273%**
Error:	**13.727%**
Cohen's kappa (k):	**0.510**

Agreement	By Chance	TruePositives	Type II error FalsePositives	Type I error FalseNegatives	TrueNegatives	Recall	Precision	Sensitivity	Specifity	F-measure
1,232	1,117	1,232	174	47	157	0.963	0.876	0.963	0.474	0.918
157	42	157	47	174	1,232	0.474	0.770	0.474	0.963	0.587
-	-	-	-	-	-	-	-	-	-	-
-	-	-	-	-	-	-	-	-	-	-
1,389	**1,159**									

Decision Tree - R considering shear types 0,1,2,3				
	Power Law	Flat	LLJ	Others
Power Law	1,165	46	54	14
Flat	41	42	1	0
LLJ	49	1	175	3
Others	16	1	1	1

Total # of subjects:	**1,610**
Correct classified:	**1,383**
Wrong classified:	**227**
Accuracy:	**85.901%**
Error:	**14.099%**
Cohen's kappa (k):	**0.597**

Agreement	By Chance	TruePositives	Type II error FalsePositives	Type I error FalseNegatives	TrueNegatives	Recall	Precision	Sensitivity	Specifity	F-measure
1,165	1,010	1,165	106	114	225	0.911	0.917	0.911	0.680	0.914
42	5	42	48	42	1,478	0.500	0.467	0.500	0.969	0.483
175	33	175	56	53	1,326	0.768	0.758	0.768	0.959	0.763
1	0	1	17	18	1,347	0.053	0.056	0.053	0.988	0.054
1,383	**1,047**									

Naive Bayes

Naïve Bayes					Total # of subjects:	1,610				Type II error	Type I error						
	Power Law	No Power Law			Correct classified:	1,188	Agreement	By Chance	TruePositives	FalsePositives	FalseNegatives	TrueNegatives	Recall	Precision	Sensitivity	Specifity	F-measure
Power Law	952	327			Wrong classified:	422	952	832	952	95	327	236	0.744	0.909	0.744	0.713	0.819
No Power Law	95	236			Accuracy:	73.789%	236	116	236	327	95	952	0.713	0.419	0.713	0.744	0.528
					Error:	26.211%	-	-	-	-	-	-	-	-	-	-	-
					Cohen's kappa (k):	0.363	-	-	-	-	-	-	-	-	-	-	-
Area Under Curve:		0.791					1,188	947									

Naïve Bayes considering shear types 0,1,2,3					Total # of subjects:	1,610				Type II error	Type I error						
	Power Law	Flat	LLJ	Others	Correct classified:	1,251	Agreement	By Chance	TruePositives	FalsePositives	FalseNegatives	TrueNegatives	Recall	Precision	Sensitivity	Specifity	F-measure
Power Law	1,059	5	205	10	Wrong classified:	359	1,059	931	1,059	113	220	218	0.828	0.904	0.828	0.659	0.864
Flat	64	6	14	0	Accuracy:	77.702%	6	1	6	5	78	1,521	0.071	0.545	0.071	0.997	0.126
LLJ	38	0	183	7	Error:	22.298%	183	58	183	224	45	1,158	0.803	0.450	0.803	0.838	0.576
Others	11	0	5	3	Cohen's kappa (k):	0.421	3	0	3	17	16	1,213	0.158	0.150	0.158	0.986	0.154
							1,251	989									

Logistic Regression

Logistic Regression					Total # of subjects:	1,610				Type II error	Type I error						
	Power Law	No Power Law			Correct classified:	1,362	Agreement	By Chance	TruePositives	FalsePositives	FalseNegatives	TrueNegatives	Recall	Precision	Sensitivity	Specifity	F-measure
Power Law	1,252	27			Wrong classified:	248	1,252	1,170	1,252	221	27	110	0.979	0.850	0.979	0.332	0.910
No Power Law	221	110			Accuracy:	84.596%	110	28	110	27	221	1,252	0.332	0.803	0.332	0.979	0.470
					Error:	15.404%	-	-	-	-	-	-	-	-	-	-	-
					Cohen's kappa (k):	0.398	-	-	-	-	-	-	-	-	-	-	-
Area Under Curve:		0.777					1,362	1,198									

Logistic Regression considering shear types 0,1,2,3					Total # of subjects:	1,610				Type II error	Type I error						
	Power Law	Flat	LLJ	Others	Correct classified:	1,377	Agreement	By Chance	TruePositives	FalsePositives	FalseNegatives	TrueNegatives	Recall	Precision	Sensitivity	Specifity	F-measure
Power Law	1,233	14	32	0	Wrong classified:	233	1,233	1,126	1,233	184	46	147	0.964	0.870	0.964	0.444	0.915
Flat	74	9	1	0	Accuracy:	85.528%	9	1	9	14	75	1,512	0.107	0.391	0.107	0.991	0.168
LLJ	93	0	135	0	Error:	14.472%	135	24	135	35	93	1,347	0.592	0.794	0.592	0.975	0.678
Others	17	0	2	0	Cohen's kappa (k):	0.492	-	-	-	-	19	1,356	-	-	-	1.000	-
							1,377	1,151									

Tree Ensemble

Tree Ensemble				
	Power Law	No Power Law		
Power Law	1,246	33		
No Power Law	105	226		
Area Under Curve:		**0.931**		

Total # of subjects:	**1,610**
Correct classified:	**1,472**
Wrong classified:	**138**
Accuracy:	**91.429%**
Error:	**8.571%**
Cohen's kappa (k):	**0.715**

Agreement	By Chance	TruePositives	Type II error FalsePositives	Type I error FalseNegatives	TrueNegatives	Recall	Precision	Sensitivity	Specifity	F-measure
1,246	1,073	1,246	105	33	226	0.974	0.922	0.974	0.683	0.948
226	53	226	33	105	1,246	0.683	0.873	0.683	0.974	0.766
-	-	-	-	-	-	-	-	-	-	-
-	-	-	-	-	-	-	-	-	-	-
1,472	**1,126**									

Tree Ensemble considering shear types 0,1,2,3				
	Power Law	Flat	LLJ	Others
Power Law	1,252	7	17	3
Flat	52	32	0	0
LLJ	44	0	184	0
Others	14	1	1	3

Total # of subjects:	**1,610**
Correct classified:	**1,471**
Wrong classified:	**139**
Accuracy:	**91.366%**
Error:	**8.634%**
Cohen's kappa (k):	**0.720**

Agreement	By Chance	TruePositives	Type II error FalsePositives	Type I error FalseNegatives	TrueNegatives	Recall	Precision	Sensitivity	Specifity	F-measure
1,252	1,082	1,252	110	27	221	0.979	0.919	0.979	0.668	0.948
32	2	32	8	52	1,518	0.381	0.800	0.381	0.995	0.516
184	29	184	18	44	1,364	0.807	0.911	0.807	0.987	0.856
3	0	3	3	16	1,451	0.158	0.500	0.158	0.998	0.240
1,471	**1,113**									

K Nearest Neighbor models.

K Nearest Neighbor				
	Power Law	No Power Law		
Power Law	1,232	47		
No Power Law	126	205		
Area Under Curve:		**0.877**		

Total # of subjects:	**1,610**
Correct classified:	**1,437**
Wrong classified:	**173**
Accuracy:	**89.255%**
Error:	**10.745%**
Cohen's kappa (k):	**0.639**

Agreement	By Chance	TruePositives	Type II error FalsePositives	Type I error FalseNegatives	TrueNegatives	Recall	Precision	Sensitivity	Specifity	F-measure
1,232	1,079	1,232	126	47	205	0.963	0.907	0.963	0.619	0.934
205	52	205	47	126	1,232	0.619	0.813	0.619	0.963	0.703
-	-	-	-	-	-	-	-	-	-	-
-	-	-	-	-	-	-	-	-	-	-
1,437	**1,131**									

K Nearest Neighbor considering shear types 0,1,2,3				
	Power Law	Flat	LLJ	Others
Power Law	1,238	9	31	1
Flat	62	21	1	0
LLJ	57	0	168	3
Others	12	1	2	4

Total # of subjects:	**1,610**
Correct classified:	**1,431**
Wrong classified:	**179**
Accuracy:	**88.882%**
Error:	**11.118%**
Cohen's kappa (k):	**0.636**

Agreement	By Chance	TruePositives	Type II error FalsePositives	Type I error FalseNegatives	TrueNegatives	Recall	Precision	Sensitivity	Specifity	F-measure
1,238	1,088	1,238	131	41	200	0.968	0.904	0.968	0.604	0.935
21	2	21	10	63	1,516	0.250	0.677	0.250	0.993	0.365
168	29	168	34	60	1,348	0.737	0.832	0.737	0.975	0.781
4	0	4	4	15	1,410	0.211	0.500	0.211	0.997	0.296
1,431	**1,118**									

End of “Results of testing the models removing correlated columns and normalizing columns before applying models.” subsection.

Summary for Part A

Results of testing the models without removing correlated columns.		
Model	**Accuracy**	**Rank**
Random Forest	**92.112%**	1
Decision Tree - R	**86.894%**	4
Naïve Bayes	**71.366%**	6
Logistic Regression	**84.534%**	5
Tree Ensemble	**92.050%**	2
K Nearest Neighbor	**90.373%**	3
Random Forest considering shear types 0,1,2,3	**92.422%**	1
Decision Tree - R considering shear types 0,1,2,3	**87.019%**	4
Naïve Bayes considering shear types 0,1,2,3	**76.522%**	6
Logistic Regression considering shear types 0,1,2,3	**85.590%**	5
Tree Ensemble considering shear types 0,1,2,3	**92.174%**	2
K Nearest Neighbor considering shear types 0,1,2,3	**90.062%**	3

Results of testing the models removing correlated columns.		
Random Forest	**91.056%**	2
Decision Tree - R	**86.273%**	4
Naïve Bayes	**73.789%**	6
Logistic Regression	**84.596%**	5
Tree Ensemble	**91.429%**	1
K Nearest Neighbor	**89.255%**	3
Random Forest considering shear types 0,1,2,3	**91.739%**	1
Decision Tree - R considering shear types 0,1,2,3	**85.901%**	4
Naïve Bayes considering shear types 0,1,2,3	**77.702%**	6
Logistic Regression considering shear types 0,1,2,3	**85.528%**	5
Tree Ensemble considering shear types 0,1,2,3	**91.366%**	2
K Nearest Neighbor considering shear types 0,1,2,3	**88.882%**	3

Results of testing the models removing correlated columns, and removing 1st and last 3 days of data.		
Random Forest	**95.747%**	2
Decision Tree - R	**95.660%**	4
Naïve Bayes	**95.486%**	6
Logistic Regression	**95.660%**	4
Tree Ensemble	**95.833%**	1

K Nearest Neighbor	**95.747%**	2
Random Forest considering shear types 0,1,2,3	**95.660%**	3
Decision Tree - R considering shear types 0,1,2,3	**93.316%**	6
Naïve Bayes considering shear types 0,1,2,3	**94.444%**	5
Logistic Regression considering shear types 0,1,2,3	**95.660%**	3
Tree Ensemble considering shear types 0,1,2,3	**95.747%**	1
K Nearest Neighbor considering shear types 0,1,2,3	**95.747%**	1

Results of testing the models removing correlated columns and LLJ data rows		
Random Forest	**94.178%**	2
Decision Tree - R	**93.086%**	4
Naïve Bayes	**92.649%**	6
Logistic Regression	**92.868%**	5
Tree Ensemble	**94.396%**	1
K Nearest Neighbor	**93.814%**	3
Random Forest considering shear types 0,1,2,3	**94.323%**	1
Decision Tree - R considering shear types 0,1,2,3	**90.684%**	6
Naïve Bayes considering shear types 0,1,2,3	**91.339%**	5
Logistic Regression considering shear types 0,1,2,3	**92.649%**	4
Tree Ensemble considering shear types 0,1,2,3	**94.250%**	2
K Nearest Neighbor considering shear types 0,1,2,3	**93.741%**	3

Results of testing the models removing correlated columns and normalizing columns before applying models.		
Random Forest	**91.056%**	2
Decision Tree - R	**86.273%**	4
Naïve Bayes	**73.789%**	6
Logistic Regression	**84.596%**	5
Tree Ensemble	**91.429%**	1
K Nearest Neighbor	**89.255%**	3
Random Forest considering shear types 0,1,2,3	**91.739%**	1
Decision Tree - R considering shear types 0,1,2,3	**85.901%**	4
Naïve Bayes considering shear types 0,1,2,3	**77.702%**	6
Logistic Regression considering shear types 0,1,2,3	**85.528%**	5
Tree Ensemble considering shear types 0,1,2,3	**91.366%**	2
K Nearest Neighbor considering shear types 0,1,2,3	**88.882%**	3

No matter the scenario tested, notice that "**Random Forest**" and "**Tree-Ensemble**" models are showing the best performance in terms of Accuracy (the proportion of the total number of predictions that were correct). These can be classified as Excellent or Good models.

It is interesting to see that "**K Nearest Neighbor**" model appears in first place in "Results of testing the models removing correlated columns, and removing 1st and last 3 days of data." scenario.

Results for Part B

This section shows the results of models applied to continuous output. Similar to Part A, the results are put into its tested scenario (see section "Model Selection node for continuous output.").

In addition, the results are showed if outliers were removed or not.

Results of testing the models removing correlated columns.

Random Forest

Removing correlated columns. (without outliers)	
Random Forest	
	Prediction
R^2	0.899
mean absolute error	0.048
mean squared error	0.004
root mean squared deviation	0.064
mean signed difference	(0.001)

Removing correlated columns. (with outliers)	
Random Forest	
	Prediction
R^2	0.913
mean absolute error	0.048
mean squared error	0.004
root mean squared deviation	0.063
mean signed difference	0.003

Tree Ensemble

Removing correlated columns. (without outliers) Tree Ensemble	Prediction
R^2	0.899
mean absolute error	0.048
mean squared error	0.004
root mean squared deviation	0.064
mean signed difference	0.000

Removing correlated columns. (with outliers) Tree Ensemble	Prediction
R^2	0.910
mean absolute error	0.048
mean squared error	0.004
root mean squared deviation	0.064
mean signed difference	0.003

Gradient Boosted Trees

Removing correlated columns. (without outliers) Gradient Boosted Trees	Prediction
R^2	0.877
mean absolute error	0.053
mean squared error	0.005
root mean squared deviation	0.071
mean signed difference	(0.001)

Removing correlated columns. (with outliers) Gradient Boosted Trees	Prediction
R^2	0.891
mean absolute error	0.053
mean squared error	0.005
root mean squared deviation	0.071
mean signed difference	0.003

Linear Regression

Removing correlated columns. (without outliers) Linear Regression	Prediction
R^2	0.817
mean absolute error	0.063
mean squared error	0.007
root mean squared deviation	0.086
mean signed difference	0.001

Removing correlated columns. (with outliers) Linear Regression	Prediction
R^2	0.804
mean absolute error	0.068
mean squared error	0.009
root mean squared deviation	0.095
mean signed difference	0.000

Polynomial Regression

Removing correlated columns. (without outliers)	
Polynomial Regression	
	Prediction
R^2	0.830
mean absolute error	0.061
mean squared error	0.007
root mean squared deviation	0.083
mean signed difference	(0.000)

Removing correlated columns. (with outliers)	
Polynomial Regression	
	Prediction
R^2	0.819
mean absolute error	0.064
mean squared error	0.008
root mean squared deviation	0.091
mean signed difference	0.001

End of “Results of testing the models removing correlated columns.” subsection.

Results of testing the models removing correlated columns, and removing 1st and last 3 days of data.

Random Forest

Removing correlated columns, and removing 1st and last 3 days of data. (without outliers)	
Random Forest	
	Prediction
R^2	0.900
mean absolute error	0.048
mean squared error	0.004
root mean squared deviation	0.063
mean signed difference	0.001

Removing correlated columns, and removing 1st and last 3 days of data. (with outliers)	
Random Forest	
	Prediction
R^2	0.901
mean absolute error	0.049
mean squared error	0.004
root mean squared deviation	0.066
mean signed difference	0.001

Tree Ensemble

Removing correlated columns, and removing 1st and last 3 days of data. (without outliers)	
Tree Ensemble	
	Prediction
R^2	0.901
mean absolute error	0.047
mean squared error	0.004
root mean squared deviation	0.063
mean signed difference	0.002

Removing correlated columns, and removing 1st and last 3 days of data. (with outliers)	
Tree Ensemble	
	Prediction
R^2	0.900
mean absolute error	0.050
mean squared error	0.004
root mean squared deviation	0.066
mean signed difference	0.001

Gradient Boosted Trees

Removing correlated columns, and removing 1st and last 3 days of data. (without outliers)	
Gradient Boosted Trees	
	Prediction
R^2	0.886
mean absolute error	0.051
mean squared error	0.004
root mean squared deviation	0.067
mean signed difference	0.001

Removing correlated columns, and removing 1st and last 3 days of data. (with outliers)	
Gradient Boosted Trees	
	Prediction
R^2	0.886
mean absolute error	0.054
mean squared error	0.005
root mean squared deviation	0.071
mean signed difference	0.001

Linear Regression

Removing correlated columns, and removing 1st and last 3 days of data. (without outliers)	
Linear Regression	
	Prediction
R^2	0.829
mean absolute error	0.063
mean squared error	0.007
root mean squared deviation	0.082
mean signed difference	0.002

Removing correlated columns, and removing 1st and last 3 days of data. (with outliers)	
Linear Regression	
	Prediction
R^2	0.803
mean absolute error	0.067
mean squared error	0.009
root mean squared deviation	0.093
mean signed difference	(0.001)

Polynomial Regression

Removing correlated columns, and removing 1st and last 3 days of data. (without outliers)	
Polynomial Regression	
	Prediction
R^2	0.835
mean absolute error	0.061
mean squared error	0.006
root mean squared deviation	0.081
mean signed difference	0.004

Removing correlated columns, and removing 1st and last 3 days of data. (with outliers)	
Polynomial Regression	
	Prediction
R^2	0.816
mean absolute error	0.064
mean squared error	0.008
root mean squared deviation	0.090
mean signed difference	0.001

End of “Results of testing the models removing correlated columns, and removing 1st and last 3 days of data.” subsection.

Summary for Part B

what is removed?	With outliers?	Model	R^2	mean absolute error	mean squared error	root mean squared deviation	mean signed difference	R^2 Rank within outliers group	R^2 Rank within correlated (with & without 4 days)
Correlated columns	No	Random Forest	0.899	0.048	0.004	0.064	(0.001)	2	4
		Tree Ensemble	0.899	0.048	0.004	0.064	0.000	1	3
		Gradient Boosted Trees	0.877	0.053	0.005	0.071	(0.001)	3	6
		Linear Regression	0.817	0.063	0.007	0.086	0.001	5	9
		Polynomial Regression	0.830	0.061	0.007	0.083	(0.000)	4	7
	Yes	Random Forest	0.913	0.048	0.004	0.063	0.003	1	1
		Tree Ensemble	0.910	0.048	0.004	0.064	0.003	2	2
		Gradient Boosted Trees	0.891	0.053	0.005	0.071	0.003	3	5
		Linear Regression	0.804	0.068	0.009	0.095	0.000	5	10
		Polynomial Regression	0.819	0.064	0.008	0.091	0.001	4	8
Correlated columns and 4 days	No	Random Forest	0.900	0.048	0.004	0.063	0.001	2	3
		Tree Ensemble	0.901	0.047	0.004	0.063	0.002	1	2
		Gradient Boosted Trees	0.886	0.051	0.004	0.067	0.001	3	5
		Linear Regression	0.829	0.063	0.007	0.082	0.002	5	8
		Polynomial Regression	0.835	0.061	0.006	0.081	0.004	4	7
	Yes	Random Forest	0.901	0.049	0.004	0.066	0.001	1	1
		Tree Ensemble	0.900	0.050	0.004	0.066	0.001	2	4
		Gradient Boosted Trees	0.886	0.054	0.005	0.071	0.001	3	6
		Linear Regression	0.803	0.067	0.009	0.093	(0.001)	5	10
		Polynomial Regression	0.816	0.064	0.008	0.090	0.001	4	9

TABLE 16 SUMMARY TABLE OF MODELS RESULTS FOR PART B

In the previous tables, the models that give better performance are the “Random Forest” and “Tree-Ensemble”, which R-Squared values are around 0.9. It indicates that the model explains ~90% the variability of the response data around its mean, this could be interpreted as **good**. Also notice that Root Mean Squared Deviation values for these models are low.

Conclusions

Concrete conclusions

General

These are general conclusions:

- There are some confounders when data is analyzed
  - E.g. the features m58, m103 and m122 are not used
- Review what happened with the data during the days 17, 26, 27 and 28
  - Few data for days 17 and 28, versus other days, that could lead us to think that those days something wrong happened, or they started to collect data at some point of day first and stop collecting data at some point of last day
  - About 79% of data is classified as "Power Law", which could dominate the analysis.
  - Days 26, 27 y 28 could be affected by the outliers of “LLJ” shear type class.
  - Looks weird that last three days of data, mostly were classified "LLJ", "Flat" or "Others", and few of them as "Power Law".
- Variables have different scales. Normalize data or remove outliers is recommended.
- Review the variance in some variables (see table "Statistics of Load Sensors Data Variables")
  - The variance could be affected by the mix of shear types.
- Review with SME if does it make sense to have a strong correlation between some X variables.
  - There are some variables with strong correlation. Variables can be reduced and just select one variable. See correlation tables above.
- Some variables are having many values around ZERO. SME (subject matter experts) need to be consulted.
  - Review if are valid to have high number of ZEROS for NODD_3C, NODD_3S, PITCH_D_0P, PITCH_Q_0P, PITCH_D_3C, PITCH_D_3S, YAW_0P, YAW_3C and PITCH_COL_0P

Answer to business requirement:

- If we only take these result to decide if GE could launch an NPI program to offer a “Virtual Wind Shear Sensor”, the answer could be yes; however, as part of a cost-benefit of this program, there are some aspects to consider in order to cut down possible risks that would put in danger this program:
  - Collect more data during more days. 12 days (from 6/17/2015 to 6/28/2015) is not too representative to all different weather conditions that are present on each season.
  - There are businesses process that could impact the results; e.g. the conditions of how is operated the wind turbine, how often is on maintenance, if communication is broken and no data is collected, transitions between months (end of month), etc. It would be interesting to see how the data behaves on other conditions.

- Review what happened during 26, 27, 28 days for data classified as “Power Law”, and clean it up.
- Cost-benefit analysis can complement this work and support or reject to do the NPI program.

Part A

Here are the conclusions for Part A.

- Mostly, no matter the scenario tested, “**Random Forest**” and “**Tree-Ensemble**” models are showing the best performance in terms of Accuracy (great than 90%).
- “Random Forest” and “Tree-Ensemble” models give better results when are removed 4 days of data (days 17, 26, 27, 28), because these days have few data for Shear Type Power Law and higher volume of data for “Flat”, “LLJ” and “Others”. **Definitely, these 4-day impact in the results.**
- “Random Forest” and “Tree-Ensemble” models give good results when “LLJ” data is removed. Remember that LLJ showed outliers for most of variables
- “Results of testing the models removing correlated columns and normalizing columns before applying models” and “Results of testing the models removing correlated columns.” show similar results, so there is no substantial impact if X variables are normalized.
- Interesting to see that “K Nearest Neighbor” model appears in first place in “Results of testing the models removing correlated columns, and removing 1st and last 3 days of data.” scenario.

Part B

These are the conclusions for Part B

- Models that give better performance are the “**Random Forest**” and “**Tree-Ensemble**”, which R-Squared values are around 0.9. It indicates that the model explains ~90% the variability of the response data around its mean, this could be interpreted as good.
- There is not too much impact in the results when outliers are removed in the case when removed correlated columns and 4 days of data. Differently from Part A, here seems there is no impact if we remove or not those 4 days (days 17, 26, 27, 28).

what is removed?	With outliers?	Model	R^2
Correlated columns and 4 days	No	Random Forest	0.900
		Tree Ensemble	0.901
		Gradient Boosted Trees	0.886
		Linear Regression	0.829
		Polynomial Regression	0.835
	Yes	Random Forest	0.901
		Tree Ensemble	0.900
		Gradient Boosted Trees	0.886
		Linear Regression	0.803
		Polynomial Regression	0.816

- However, there is a small impact when outliers are removed in the case when correlated columns are removed, but are not removed those 4 days.

what is removed?	With outliers?	Model	R^2
Correlated columns	No	Random Forest	0.899
		Tree Ensemble	0.899
		Gradient Boosted Trees	0.877
		Linear Regression	0.817
		Polynomial Regression	0.830
	Yes	Random Forest	0.913
		Tree Ensemble	0.910
		Gradient Boosted Trees	0.891
		Linear Regression	0.804
		Polynomial Regression	0.819

- So, it is important to review again what happened during those 4 days (days 17, 26, 27, 28).

Part C

Conclusions for Part C

- **Consult to SME** about how to calculate the shear when speed profile does not follow a power law. See Part C goal section above.
- Depending on the **new equation** for each speed profile when ShearTypeClass is: 1 = LLJ, 2 = Flat, and 3 = Others, the process followed to solve Part B goal can be **adjusted** to deal with these cases.
- Depending on what SME will suggest, maybe we will need to **extrapolate** from how was calculated the shear when it follows the power law, to when it does not follow.
- As reviewed in section “Exploration of data” (above in this document). There are few data for the types 1, 2 and 3 in the dataset provided (the total number of rows for these 3 types is 1,700 records). Strongly recommended to collect more data.
- If SMEs suggest that is enough these 3 days of data for types 1, 2 and 3 (1,436 records of 1,700 were collected in 3 days for these 3 types), then we need to re-evaluate the models applied, and find out if there are another models that work better with few data.
- In general, to deal with PART C, guidance from SMEs is requested.

Potential problems with the conclusions

According to literature consulted and the analysis performed, there are some points to consider when the results were interpreted, and conclusions were reached out.

- Obtained results (models that showed better performance) could be different if more data is collected.
- Data was collected during ending spring and beginning summer. Most probably, the behavior of data collected could be different versus other seasons. So, more representative data would be great to have it.
- AuC or Accuracy is not always enough to validate the models.
- AUC for ordinal regression is something tricky. You might want to calculate the AUC for each class by creating dummies to take value 1 for the class you.[7]. For these reason was included the scenario to have as main target class the shear type "Power Law" and the rest classified as "No Power Law".
- Accuracy often is not sufficient/appropriate[8]
    - Assumes equal cost for both kinds of errors
        - cost(b-type-error) = cost (c-type-error)
    - is 99% accuracy good?
        - can be excellent, good, mediocre, poor, terrible
        - depends on problem
- For Part B, R-Squared interpretation could be not enough. Take care of R-Squared interpretation, depending on its context. See "What's a good value for R-squared?"[9] publication.
- R-Squared only gives you information about "goodness of fit", not "goodness of prediction" which is far more useful in practice.[10]
- As mentioned before, there are values close to ZERO in different variables, and this could produce a lot of noise, so the models could be "over-fitted". Although, it is mentioned Random Forest arguably does not overfit.

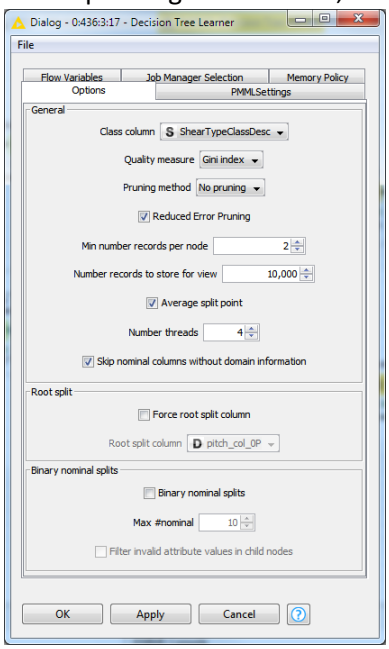
Further validation and/or next steps

Before to start the NPI program to offer a "Virtual Wind Shear Sensor", consider these next steps to have better certainty:

- As mentioned above, collect more data
- Consult SME for all questions or concerns mentioned above
- Review if models are not over-fitted.
- Algorithms Tuning. The algorithms can produce different results depending on how they are parametrized. E.g. notice in the following picture, the Decision Tree algorithm has several parameters,

[7] AUC in Ordinal Logistic Regression. http://stats.stackexchange.com/questions/23992/auc-in-ordinal-logistic-regression
[8] Performance Measures for Machine Learning. http://www.cs.cornell.edu/courses/cs678/2006sp/performance_measures.4up.pdf
[9] What's a good value for R-squared? http://people.duke.edu/~rnau/rsquared.htm
[10] What is an acceptable range for r-squared in real-world environments? https://www.quora.com/What-is-an-acceptable-range-for-r-squared-in-real-world-environments

and depending on its values, the algorithm can give better or worst performance.



- Creating models is an iterative process, so if we want to have a solid NPI, a well-established process is required to collect new data, create models, run models again and compare the results.

Takeaways.

Additional to conclusions and next steps, there are some lessons learnt during the analysis of this case study. Here are some takeaways:

- There is a potential risk we could stay analyzing and analyzing models for a long time and not get conclusions.
- There is no a unique solution.
- Garbage in, garbage out. It is very important cleaning the data in the first place.
- The process followed to get these results, could be slow and cumbersome due to the performance of models, new findings, new models, tuning models, getting different results on each run, etc.
- Definitely, tools are very useful to do this kind of analysis. Tools like KNIME, Tableau, R, Python, Octave, etc. I could say, there is no better or worst tool, depends on how many algorithms provide, if algorithms are solid, if tool is supported by wide audience, there is a documentation, examples, demos, etc.
- These tools help to compare different machine learning models to quickly identify the best one. Also, help to automated ensemble model evaluation to identify the best performers.
- Before doing this case study, I didn't know about Wind Turbine principles, terms, components, etc. Now, I know much more, but little compared to all technology that is behind.
- In addition, it was very illustrative to me doing this case study because it was more clear to me how to use classification and predictive models, using a real example.
- Also, I learnt a lot about KNIME, R and Tableau tools for analysis and modeling.

Glossary

Term	Definition
Area Under the Curve (AuC)	Area Under Curve, is a metric for binary classification. http://fastml.com/what-you-wanted-to-know-about-auc/ http://machinelearningmastery.com/assessing-comparing-classifier-performance-roc-curves-2/
Low-level Jet (LLJ)	Low-level Jet (LLJ) is major warm-season wind resource in the U.S. Great Plains and it plays an important role in generating shear and turbulence within the atmospheric layer occupied by wind turbine rotors. http://www.esrl.noaa.gov/csd/projects/lamar/llj.html The development of a wind speed maximum in the nocturnal boundary layer, referred to as a low-level jet (LLJ), is a common feature of the vertical structure of the atmospheric boundary layer (ABL). https://www.researchgate.net/publication/241300242_CFD_modelling_of_nocturnal_low-level_jet_effects_on_wind_energy_related_variables
Random Forest	A Random Forest consists of a collection or ensemble of simple tree predictors, each capable of producing a response when presented with a set of predictor values. For classification problems, this response takes the form of a class membership, which associates, or classifies, a set of independent predictor values with one of the categories present in the dependent variable. Alternatively, for regression problems, the tree response is an estimate of the dependent variable given the predictors. The Random Forest algorithm was developed by Breiman. https://www.statsoft.com/Textbook/Random-Forest
Roughness	Surface roughness often shortened to roughness, is a component of surface texture. It is quantified by the deviations in the direction of the normal vector of a real surface from its ideal form. If these deviations are large, the surface is rough; if they are small, the surface is smooth. https://en.wikipedia.org/wiki/Surface_roughness
RPM	Rotations Per Minute
R-Squared	R-squared is the “percent of variance explained” by the model. That is, R-squared is the fraction by which the variance of the errors is less than the variance of the dependent variable. http://people.duke.edu/~rnau/rsquared.htm http://www.theanalysisfactor.com/assessing-the-fit-of-regression-models/
Tree Ensembles	Ensemble Method combines multiple models in a certain way to fit the training data. There are two primary ways: “bagging” and “boosting”. http://horicky.blogspot.com/2012/06/predictive-analytics-decision-tree-and.html?_sm_au_=i4V6QHfJ4sL8RM6r
Wind profile power law	The wind profile power law is a relationship between the wind speeds at one height, and those at another. https://en.wikipedia.org/wiki/Wind_profile_power_law
Wind Shear	Wind shear is the difference in wind speed by height. The higher the wind shear, the higher the wind speeds aloft when winds are close to calm on the ground. https://windwisema.org/wind_shear-turbine_noise-faq/

References / further reading in general

Here are some good references to delve into machine learning algorithms.

Ethem Alpaydın (2014). *Introduction to Machine Learning – Third Edition*. The MIT Press.

Gareth James, Daniela Witten, Trevor Hastie, Robert Tibshirani. (2013). *An Introduction to Statistical Learning with Applications in R*. Springer.

Peter Harrington (2012). *Machine Learning in Action*. Manning Publications.

Shai Shalev-Shwartz and Shai Ben-David (2014). *Understanding Machine Learning: From Theory to Algorithms*. Cambridge University Press.

Trevor Hastie, Robert Tibshirani, Jerome Friedman (2008). *The Elements of Statistical Learning – Data Mining, Inference, and Prediction. Second Edition*. Springer Series in Statistics.

Appendix

What to consider when choosing a predictive or classification model?

It could be a hard task to find out where to start, finding the right models, what models to consider, are models liked to a specific problem? How to know if they fit with the problem to deal with, etc. There are different recourses that are good starting point. Next sections will mention a couple.

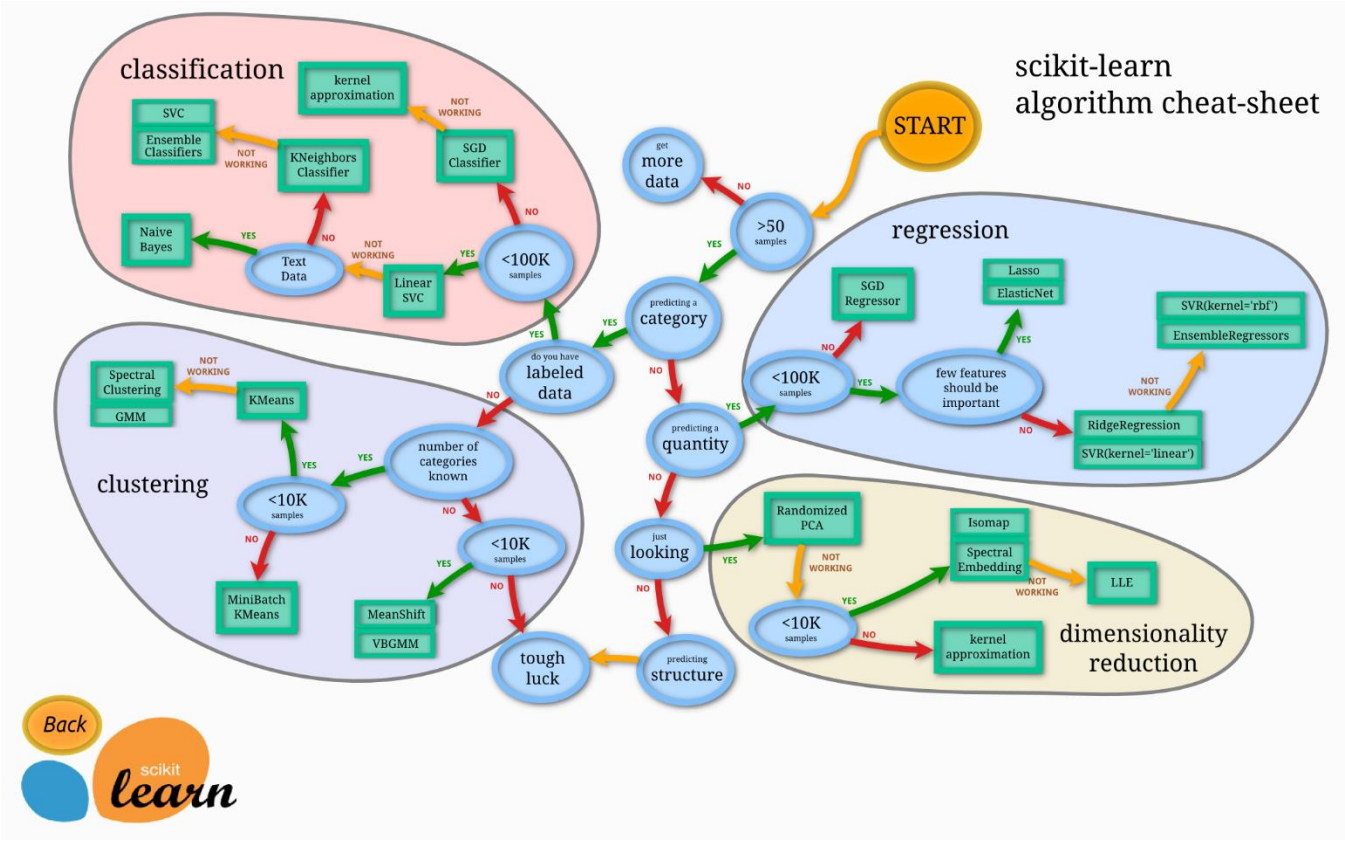
What factors should I consider when choosing a predictive model technique?[11]

Here are some suggestions:

1. How does your target variable look like?
    a. continuous target variable? -> regression
    b. categorical (nominal) target variable? -> classification
    c. ordinal target variable? -> ranked classification
    d. no target variable and want to find structure in data? -> cluster analysis, projection
2. Is computational performance an issue?
    a. use “cheaper” models/algorithms
    b. dimensionality reduction
    c. feature selection
    d. lazy learner (e.g,. k-nearest neighbors)
3. Does my dataset fit into memory?
    a. out of core learning
    b. distributed systems
4. Is my data linearly separable?
    a. hard to know the answer upfront
    b. always a good idea to compare different models
5. Finding a good bias variance threshold. Does my model overfit?
    a. increase regularization strength if supported by the model
    b. dimensionality reduction or feature selection otherwise
    c. collect more training data if possible (check via learning curves first)
6. Are you planning to update your model with new data on the fly?
    a. one option are lazy learners (e.g., K-nearest neighbors); needs to keep training data around; no learning necessary but more expensive predictions
    b. it’s generally relatively cheap to update generative models
    c. another option is stochastic gradient descent for online learning

[11] What factors should I consider when choosing a predictive model technique?
http://sebastianraschka.com/faq/docs/choosing-technique.html
https://www.quora.com/What-factors-should-I-consider-when-choosing-a-predictive-model-technique

See next picture, it is a good starting point[12]:



A Tour of Machine Learning Algorithms[13]

It is another good lecture to take a look of the most popular algorithms. It explains different learning styles in machine learning algorithms, such as:

- Supervised Learning
- Unsupervised Learning
- Semi-Supervised Learning

Also, this article list different algorithms which are grouped by similarity in terms of their function (how they work). For example, tree-based methods, and neural network inspired methods.

Regression Algorithms

The most popular regression algorithms are:

[12] Machine Learning Map. http://scikit-learn.org/dev/tutorial/machine_learning_map/index.html
[13] A Tour of Machine Learning Algorithms. http://machinelearningmastery.com/a-tour-of-machine-learning-algorithms/

- Ordinary Least Squares Regression (OLSR)
- Linear Regression
- Logistic Regression
- Stepwise Regression
- Multivariate Adaptive Regression Splines (MARS)
- Locally Estimated Scatterplot Smoothing (LOESS)

Instance-based Algorithms

The most popular instance-based algorithms are:

- k-Nearest Neighbour (kNN)
- Learning Vector Quantization (LVQ)
- Self-Organizing Map (SOM)
- Locally Weighted Learning (LWL)

Regularization Algorithms

The most popular regularization algorithms are:

- Ridge Regression
- Least Absolute Shrinkage and Selection Operator (LASSO)
- Elastic Net
- Least-Angle Regression (LARS)

Decision Tree Algorithms

The most popular decision tree algorithms are:

- Classification and Regression Tree (CART)
- Iterative Dichotomiser 3 (ID3)
- C4.5 and C5.0 (different versions of a powerful approach)
- Chi-squared Automatic Interaction Detection (CHAID)
- Decision Stump
- M5
- Conditional Decision Trees

Bayesian Algorithms

The most popular Bayesian algorithms are:

- Naive Bayes
- Gaussian Naive Bayes
- Multinomial Naive Bayes
- Averaged One-Dependence Estimators (AODE)
- Bayesian Belief Network (BBN)
- Bayesian Network (BN)

Clustering Algorithms

The most popular clustering algorithms are:

- k-Means
- k-Medians
- Expectation Maximisation (EM)
- Hierarchical Clustering

Association Rule Learning Algorithms

The most popular association rule learning algorithms are:

- Apriori algorithm
- Eclat algorithm

Artificial Neural Network Algorithms

The most popular artificial neural network algorithms are:

- Perceptron
- Back-Propagation
- Hopfield Network
- Radial Basis Function Network (RBFN)

Deep Learning Algorithms

The most popular deep learning algorithms are:

- Deep Boltzmann Machine (DBM)
- Deep Belief Networks (DBN)
- Convolutional Neural Network (CNN)
- Stacked Auto-Encoders

Dimensionality Reduction Algorithms

This can be useful to visualize dimensional data or to simplify data which can then be used in a supervised learning method. Many of these methods can be adapted for use in classification and regression.

- Principal Component Analysis (PCA)
- Principal Component Regression (PCR)
- Partial Least Squares Regression (PLSR)
- Sammon Mapping
- Multidimensional Scaling (MDS)
- Projection Pursuit
- Linear Discriminant Analysis (LDA)
- Mixture Discriminant Analysis (MDA)
- Quadratic Discriminant Analysis (QDA)
- Flexible Discriminant Analysis (FDA)

Ensemble Algorithms

Much effort is put into what types of weak learners to combine and the ways in which to combine them. This is a very powerful class of techniques and as such is very popular.

- Boosting
- Bootstrapped Aggregation (Bagging)
- AdaBoost
- Stacked Generalization (blending)
- Gradient Boosting Machines (GBM)
- Gradient Boosted Regression Trees (GBRT)
- Random Forest

Top 10 Machine Learning Algorithms[14]

This article lists the top machine learning and data mining algorithms. Click on link mentioned in the reference.

Microsoft Machine Learning Algorithm Cheat Sheet

If you want to know more about Microsoft ML Azure learning algorithms, take a look this document. It shows in one page the algorithms and how they are classified to be used as it may apply.

microsoft-machine-learning-algorithm-c

[14] Top 10 Machine Learning Algorithms.
http://www.datasciencecentral.com/profiles/blogs/top-10-machine-learning-algorithms