folder/uio-fp-navn-eng.pdf

teleskopgutt2.jpeg

Master's thesis

A novel application of machine learning to develop pointing models for current and future radio/sub-millimeter telescopes

Bendik Nyheim

Computational Science: Physics
60 ECTS study points

Department of Physics
Faculty of Mathematics and Natural Sciences

folder/uio-fp

Spring 2023

Bendik Nyheim

A novel application of machine learning to develop pointing models for current and future radio/sub-millimeter telescopes

Supervisors:
Signe Riemer Sørensen (Sintef)
Rodrigo Parrar (ESO)
Claudia Cicone (UiO)

Contents

Chapter 1

Introduction

Chapter 2

Related Works

The application of machine learning in astronomy has become increasingly popular in recent years, with various applications such as data analysis and prediction. However, the use of machine learning in the context of pointing models in radio telescopes has yet to be extensively explored. In this section, we provide a review of the existing literature on pointing models in radio telescopes, as well as the potential use of machine learning for similar applications.

2.1 Pointing Models in Radio Telescopes

Traditional methods for pointing models in radio telescopes involve modeling the pointing error as a function of various parameters, such as azimuth, elevation, temperature, and time. These models are often complex and require significant effort to develop and maintain. Moreover, they can be limited by the accuracy of the models used for atmospheric refraction, instrumental error, and other sources of noise.

Several papers have described various approaches to improve the pointing accuracy of radio telescopes. For example, White et al. [4] developed a pointing model for the Green Bank Telescope using theoretical terms based on the telescope's structure and analysis on the thermal deformation of the telescope structure. Greve et al. [2] studied seasonal effects on the pointing.

2.2 Machine Learning in Astronomy

Machine learning is used in various ways in astronomy. For instance, Petrillo et al. [3] used two convolutional neural networks to detect gravitational lensing from images. George & Huerta [1] used a convolutional neural network to detect gravitational waves in real time at LIGO. Despite many use cases for machine learning in astronomy and the need for an accurate pointing model in radio telescopes, we have not found any studies that used machine learning to develop or maintain a pointing model for radio telescopes.

2.3 Challenges and Opportunities

The use of machine learning for pointing models in radio telescopes poses several challenges and opportunities. One of the main challenges is the need for large datasets, which can be difficult to obtain in the context of radio telescopes. Moreover, the accuracy of the pointing model depends on the accuracy of the data used for training, which can be affected by various sources of noise and error. Nonetheless, machine learning algorithms offer the potential for significant improvements in pointing accuracy, and can potentially reduce the complexity and maintenance requirements of traditional pointing models. Future research in this area could explore the development of machine learning algorithms that can handle the challenges unique to radio telescopes, and the integration of machine learning techniques into existing pointing models.

Chapter 3

Atronomical Background

Chapter 4

Machine Learning Background

Chapter 5

Method

Chapter 6

Results

6.1 Experiment 1: Pointing Model using Neural Networks

6.2 Experiment 2: Pointing Correction Model

Table 6.1 and 6.2 show the main results from experiment 2. They contain the average compared RMS (**??**) for azimuth and elevation models in case 1 and 2 for different numbers of features k used to train the models. We cleaned the datasets we used to train these models using the cleaning criteria and XGBoost classifier. Table 6.1 shows results for models trained on scans from all instruments, while Table 6.2 shows results for models trained on scans from NFLASH230 only. For both datasets, case 1 does not offer any improvement at all. On the other hand, models from case 2 improve the pointing accuracy for all numbers of selected features k.

First, the results from models predicting offsets for all instruments. Adding complexity seems to worsen the performance of azimuth models while improving the performance of elevation models. The best performance for azimuth models is for $k = 5$ with $\bar{r}_{RMS} = 0.955$, although $k = 2$ and $k = 10$ offer similar performance. For elevation models, the best performance is for $k = 50$ with $\bar{r}_{RMS} = 0.947$.

Second, the results from models predicting offsets for NFLASH230 only. Adding complexity for azimuth does not seem to affect it in the same way, and the results are more similar for different values of k. The best result for azimuth is $k = 5$ with $\bar{r}_{RMS} = 0.917$. For elevation models, $k = 5$ also offers the best performance, with $\bar{r}_{RMS} = 0.937$.

We also conducted the same experiment for two other datasets, with the offsets and pointing corrections transformed to simulate a pointing correction after every scan. We tested splitting the training and validation data completely randomly for all four datasets, unlike the presented results, where we dedicate a day to either training or validation. For these results, see Appendix A (8).

Table 6.1: Resulting RMS from Case 1 and 2 for XGBoost model predicting pointing offset. The dataset used to get these results contain all scans and is cleaned using the regular criteria and the XGBoost classifier. The training and validation data is split on days, meaning that all the scans for a given day are in the training or validation set and not both. The test set is unaffected by this.

	RMS Case 1		RMS Case 2	
k	Azimuth	Elevation	Azimuth	Elevation
2	1.007	1.232	0.956	0.958
5	1.003	1.170	0.955	0.967
10	1.288	1.116	0.958	0.962
20	1.580	1.121	0.969	0.958
30	1.606	1.107	0.967	0.971
40	1.528	1.068	0.980	0.951
50	1.758	1.061	0.993	0.947

Table 6.2: Resulting RMS from Case 1 and 2 for XGBoost model predicting pointing offset. The dataset used to get these results contain only NFLASH230 and is cleaned using the regular criteria and the XGBoost classifier. The training and validation data is split on days, meaning that all the scans for a given day are in the training or validation set and not both. The test set is unaffected by this.

	RMS Case 1		RMS Case 2	
k	Azimuth	Elevation	Azimuth	Elevation
2	0.982	1.020	0.930	0.957
5	1.366	1.198	0.917	0.937
10	1.383	1.155	0.951	0.943
20	1.252	1.126	0.940	0.942
30	1.335	1.094	0.928	0.939
40	1.146	1.058	0.936	0.955
50	1.202	1.062	0.937	0.942

6.3 Testing

RMS Test		RMS Val		activation	hidden_layers	learning_rate	batch_size	loss_func
mean	std	mean	std	first	first	first	first	first
16.61	7.63	13.20	1.82	tanh	[[26]]	0.00	358	¡functio
18.49	7.25	15.54	1.11	tanh	[[49, 40]]	0.02	320	¡functio
18.87	11.55	14.74	2.33	relu	[[117, 32]]	0.02	298	¡functio
19.26	8.62	13.95	1.90	relu	[[83, 32]]	0.01	331	¡functio
19.44	12.50	15.73	3.05	relu	[[97, 60]]	0.02	359	¡functio

Chapter 7

Discussion

Chapter 8

Conclusion

Appendices

Appendix A

.1 Pointing Correction Results

Table 1: Resulting RMS from Case 1 and 2 for XGBoost model predicting pointing offset. The dataset used to get these results contains only data from NFLASH230 and is cleaned using the regular criteria and XGBoost classifier.

	RMS Case 1		RMS Case 2	
k	Azimuth	Elevation	Azimuth	Elevation
2	0.983957	1.018378	0.926561	0.991728
5	1.297720	1.137682	0.938401	0.987364
10	1.428557	1.153950	0.966462	1.002820
20	1.229440	1.103770	0.943179	0.984103
30	1.387079	1.064364	0.958910	0.995761
40	1.224572	1.054826	0.962620	1.004978
50	1.179139	1.073336	0.974661	0.968089

Table 2: Resulting RMS from Case 1 and 2 for XGBoost model predicting pointing offset. The dataset used to get these results contains only data from NFLASH230 and is cleaned using the regular criteria and XGBoost classifier. are transformed to simulate a pointing correction after every scan.

	RMS Case 1		RMS Case 2	
k	Azimuth	Elevation	Azimuth	Elevation
2	1.063771	1.097368	0.982346	1.023116
5	1.158891	1.135054	0.974600	1.005169
10	1.110609	1.074162	0.989497	1.005604
20	1.125938	1.001437	0.970074	1.013085
30	1.126540	1.023054	0.994982	0.971606
40	1.134314	1.065193	1.013492	1.041213
50	1.112269	1.028447	0.980788	0.998984

Table 3: Resulting RMS from Case 1 and 2 for XGBoost model predicting pointing offset. The dataset used to get these results contain all scans and is cleaned using the regular criteria and the XGBoost classifier.

	RMS Case 1		RMS Case 2	
k	Azimuth	Elevation	Azimuth	Elevation
2	1.005252	1.232152	0.963643	0.966159
5	1.002968	1.228990	0.962584	0.952695
10	1.289085	1.212770	0.969236	0.983188
20	1.474769	1.223286	0.973215	0.957379
30	1.420438	1.217082	0.977044	0.957553
40	1.359452	1.250011	0.981884	0.939400
50	1.447718	1.239426	0.960710	0.950992

Table 4: Resulting RMS from Case 1 and 2 for XGBoost model predicting pointing offset. The dataset used to get these results contain all scans and is cleaned using the regular criteria and the XGBoost classifier. It is also transformed to simulate a pointing correction after every scan.

	RMS Case 1		RMS Case 2	
k	Azimuth	Elevation	Azimuth	Elevation
2	1.411340	1.072811	0.958876	0.944501
5	1.658062	1.095572	0.959785	0.959705
10	1.656576	1.144753	0.977446	0.943979
20	1.744980	1.020705	0.974366	0.946778
30	1.891196	1.006098	0.998285	0.940884
40	2.040398	0.993940	0.972515	0.952763
50	2.226176	0.997667	0.976921	0.942357

.2 Raw data correlation

.3 Base pointing model results

Table 5: Resulting RMS from Case 1 and 2 for XGBoost model predicting pointing offset. The dataset used to get these results contain only NFLASH230 and is cleaned using the regular criteria and the XGBoost classifier. The training and validation data is split on days, meaning that all the scans for a given day are in the training or validation set and not both. The test set is unaffected by this.

	RMS Case 1		RMS Case 2	
k	Azimuth	Elevation	Azimuth	Elevation
2	0.982285	1.019755	0.929712	0.956510
5	1.365704	1.198440	0.917298	0.937182
10	1.383077	1.155258	0.951214	0.943254
20	1.251918	1.126138	0.939914	0.942015
30	1.334676	1.093938	0.928257	0.939238
40	1.145716	1.057689	0.936151	0.955285
50	1.201661	1.061612	0.937422	0.941564

Table 6: Resulting RMS from Case 1 and 2 for XGBoost model predicting pointing offset. The dataset used to get these results contain only NFLASH230 and is cleaned using the regular criteria and the XGBoost classifier. It is also transformed to simulate a pointing correction after every scan. $tmp2022\_clean\_clf\_nflash230\_transformed\_results\_table$, The training and validation data is split on days, meaning that all the scans for a given day are in the training or validation set and not both. The test set is unaffected by this.

	RMS Case 1		RMS Case 2	
k	Azimuth	Elevation	Azimuth	Elevation
2	1.030842	1.005132	0.982900	1.057857
5	1.036558	1.000412	0.983725	1.031629
10	1.136567	1.008504	0.987074	1.012463
20	1.042592	1.008786	0.977380	0.994046
30	1.242498	1.004090	0.977887	1.008825
40	1.214723	1.006611	1.006974	0.996165
50	1.180379	0.992092	0.983166	0.976880

Table 7: Resulting RMS from Case 1 and 2 for XGBoost model predicting pointing offset. The dataset used to get these results contain all scans and is cleaned using the regular criteria and the XGBoost classifier. The training and validation data is split on days, meaning that all the scans for a given day are in the training or validation set and not both. The test set is unaffected by this.

	RMS Case 1		RMS Case 2	
k	Azimuth	Elevation	Azimuth	Elevation
2	1.007153	1.232004	0.959771	0.957618
5	1.002580	1.169598	0.955311	0.967055
10	1.287770	1.115745	0.957866	0.962132
20	1.579776	1.120837	0.969443	0.957544
30	1.606464	1.107195	0.967436	0.971148
40	1.528256	1.068496	0.979982	0.951577
50	1.757519	1.061275	0.993239	0.947433

Table 8: Resulting RMS from Case 1 and 2 for XGBoost model predicting pointing offset. The dataset used to get these results contain all scans and is cleaned using the regular criteria and the XGBoost classifier. It is also transformed to simulate a pointing correction after every scan. The training and validation data is split on days, meaning that all the scans for a given day are in the training or validation set and not both. The test set is unaffected by this.

	RMS Case 1		RMS Case 2	
k	Azimuth	Elevation	Azimuth	Elevation
2	1.362174	1.049563	0.958867	0.949895
5	1.519794	1.073077	0.968773	0.951299
10	1.515574	1.043983	0.974003	0.948826
20	1.698061	1.017496	0.981890	0.939138
30	1.716675	1.014389	0.979207	0.957982
40	2.476614	1.034294	0.990171	0.947986
50	2.493252	1.031271	0.968949	0.943721

Table 9: Features with Spearman's rank correlation ≥ 0.1 to either one of the target values.

Feature	δ_{Az}	δ_{El}	$\delta_{\mathrm{Az}} \cos \mathrm{El}$
WINDSPEED_VAR_5	0.10	0.12	0.11
DAZ_TILT_MEDIAN_1	0.09	0.03	0.14
DAZ_TILTTEMP_MEDIAN_1	0.00	-0.06	0.30
TILT1Y_MEDIAN_1	0.08	0.02	0.13
TEMP26_MEDIAN_1	0.06	0.13	−0.19
TEMP27_MEDIAN_1	0.06	0.13	−0.21
TEMP28_MEDIAN_1	0.04	0.11	−0.25
TEMPERATURE_MEDIAN_1	0.04	0.12	−0.26
POSITIONZ_MEDIAN_1	0.05	0.11	−0.21
DEWPOINT_MEDIAN_1	0.08	0.12	−0.15
DAZ_TOTAL_MEDIAN_1	0.09	0.03	0.13
WINDSPEED_MEDIAN_1	0.02	0.02	0.13
HESE2	0.46	0.44	0.44
HESE3	0.98	0.94	0.77
HESE4	0.97	0.92	0.74
HESE5	0.34	0.31	0.15
HACA5	0.06	0.05	0.10
HECE	1.00	0.96	0.78
HECE2	1.00	0.96	0.78
HECE3	0.76	0.72	0.52
HSCA5	0.12	0.11	0.14

Table 10: Features with Pearson's correlation ≥ 0.1 to either one of the target values.

Feature	δ_{Az}	δ_{El}	$\delta_{\mathrm{Az}} \cos \mathrm{El}$
TEMP26_MEDIAN_1	0.05	0.11	−0.12
TEMP27_MEDIAN_1	0.05	0.11	−0.13
DAZ_TOTAL_MEDIAN_1	0.12	0.08	0.07
WINDSPEED_MEDIAN_1	0.03	0.03	0.10
AWAZ	0.00	−0.01	0.57
HESE2	0.56	0.54	0.45
HESE3	0.97	0.90	0.43
HESE4	0.91	0.82	0.26
HESE5	0.29	0.22	−0.11
HECE	1.00	0.92	0.45
HECE2	0.98	0.89	0.38
HECE3	0.70	0.62	0.10
HSCA	0.01	−0.00	0.53

Table 11: Regular neural network

Hyperparameters					Test RMS	
Activation	Hidden layers	Learning rate	Batch size	Loss	Mean[$''$]	STD
ReLU	[82]	2.0e-02	334	MSE	17.74	5.68
Tanh	[22, 46]	6.7e-03	475	MSE	17.86	10.70
GeLU	[46]	1.3e-02	472	MSD	17.91	6.79
Tanh	[64]	5.7e-03	61	MSE	18.69	8.68
ReLU	[20]	3.8e-03	309	MSD	21.42	7.22

Table 12: Separate 1

Hyperparameters					Test RMS	
Activation	Hidden layers	Learning rate	Batch size	Loss	Mean[$''$]	STD
Tanh	[40]	9.8e-03	101	MSD	16.80	7.20
GeLU	[86, 106]	1.9e-02	43	MSD	18.24	9.06
Tanh	[32]	1.7e-02	266	MSE	20.13	8.22
ReLU	[68, 42]	2.0e-02	450	MSD	20.77	9.54
ReLU	[29]	1.3e-02	148	MSD	21.60	13.01

Table 13: Separate 2

Hyperparameters					Test RMS	
Activation	Hidden layers	Learning rate	Batch size	Loss	Mean[$''$]	STD
Tanh	[40]	9.8e-03	101	MSD	19.21	5.89
Tanh	[62]	4.2e-03	155	MSD	19.77	5.93
GeLU	[54]	2.0e-02	413	MSD	21.38	6.58
GeLU	[97, 24]	1.9e-02	321	MSE	22.50	13.70
ReLU	[81]	1.2e-02	439	MSD	22.61	14.51

Table 14: Separate 3

Hyperparameters					Test RMS	
Activation	Hidden layers	Learning rate	Batch size	Loss	Mean[$''$]	STD
Tanh	[26]	3.9e-03	358	MSD	16.61	7.63
Tanh	[49, 40]	1.6e-02	320	MSD	18.49	7.25
ReLU	[117, 32]	2.1e-02	298	MSD	18.87	11.55
ReLU	[83, 32]	1.1e-02	331	MSD	19.26	8.62
ReLU	[97, 60]	1.6e-02	359	MSD	19.44	12.50

Table 15: RMS on all folds for the best model for all arcitechtures

Network	RMS on fold 1	2	3	4	5	6	Mean	STD
Regular	28.06	19.34	12.28	13.25	17.19	16.33	17.74	5.19
Sep 1	30.69	16.93	13.76	10.04	15.77	13.61	16.80	6.57
Sep 2	27.34	20.75	12.65	24.17	13.64	16.69	19.21	5.38
Sep 3	30.27	20.59	14.01	10.76	13.67	10.34	16.61	6.97

Bibliography

[1] Daniel George and Eliu Huerta. Deep learning for real-time gravitational wave detection and parameter estimation with advanced ligo data. 11 2017.

[2] A Greve, J-F Panis, and C Thum. The pointing of the iram 30-m telescope. *Astronomy and Astrophysics Supplement, v. 115, p. 379*, 115:379, 1996.

[3] C E Petrillo, C Tortora, G Vernardos, L V E Koopmans, G Verdoes Kleijn, M Bilicki, N R Napolitano, S Chatterjee, G Covone, A Dvornik, T Erben, F Getman, B Giblin, C Heymans, J T A de Jong, K Kuijken, P Schneider, H Shan, C Spiniello, and A H Wright. LinKS: discovering galaxy-scale strong lenses in the Kilo-Degree Survey using convolutional neural networks. *Monthly Notices of the Royal Astronomical Society*, 484(3):3879–3896, January 2019. _eprint: https://academic.oup.com/mnras/article-pdf/484/3/3879/28572367/stz189.pdf.

[4] E. White, F. D. Ghigo, R. M. Prestage, D. T. Frayer, R. J. Maddalena, P. T. Wallace, J. J. Brandt, D. Egan, J. D. Nelson, and J. Ray. Green Bank Telescope: Overview and analysis of metrology systems and pointing performance. *Astronomy & Astrophysics*, 659:A113, March 2022.