FYS-STK4155 - Project 3
Is shared socioeconomic pathways separable? A classification problem

Johannes Fjeldså

May 9, 2024

Abstract

Mitigation of climate change is at large decided by government and international policies. This report examins the classification of climate model realizations under Shared Socio-economic Pathway SSP126 and SSP585 using machine learning. This is an effort to efficiently support decision making by allowing for early detection of policy effectiveness. In order to count as a sucsessfull classification f1 score of 0.8 is to be acchived. Throughout I highlight the need to apply more complex models and increasing the sample size in order to give consistent and confident results. For this binary classification problem the support vector classifier emerged as the most promising classifier displaying the ability to correctly classify unseen realizations as early as 2027-2033.

1 Introduction

Climate change is a pressing issue that requires immediate and sustained attention. Policies and frameworks are being developed worldwide to mitigate its effects and adapt to its impacts. In order to imagine a future world Shared Socio-economic Pathways (SSPs) are used as a represents a different development trejectories. The SSPs vary in their assumptions on global population growth, access to education, urbanization, economic growth, resources availability, technology developments, and lifestyle changes. Climate policies play a significant role in shaping these pathways. They can influence the trajectory of greenhouse gas emissions, the pace of technological innovation, and the speed of societal transition towards sustainability.

This report presents a classification of realizations from climate models runned on different Shared Socio-economic Pathways (SSPs). The classification is performed on three-year temporal cross-sections of the climate model realizations, examining their mean values. The performance of the classification is evaluated using the F1 score, a measure of a test's accuracy that considers both precision and recall. The target F1 score for successful classification is set at 0.8. The classifiers used in this study include Support Vector Classifier (SVC), Random Forest (RF), and Decision Tree (DT).

The goal of this study is to determine how early successful classification (F1 score = 0.8) can be achieved. This will be particularly important for assessment of climate policies through out the green shift.

2 Theory

2.1 About classification problems

One of the most central problems solved by machine learning (ML) algorithms is the classification of observations. The task solved is mapping observations to discrete values or categories. For an arbitrary observation x and classifier f we have that

$$f(x) \mapsto \{0, 1\} \tag{1}$$

where $\mapsto$ denotes "maps to", and $\{0, 1\}$ will be clusters in the feature space [1].

For data that is assumed to be bimodal equation (1) is adequate to describe the relation between classifier, observation and clusters. In more complex, multi modal cases, (1) is not directly applicable, however the usage of one hot vectors will in principal reduce the mapping to equation (1) [2].

Taking a step back, and investigating the broader picture of ML I cite Goodfellow et al:

"*The factors determining how well a machine learning algorithm will perform are its ability to*

1. *Make the training error small.*
2. *Make the gap between the training and test error small.*

" [1]

The illustration in figure 1 concisely represents how one as practitioners will balance goal 1 and 2. Moving from left to right the figure refeers to; underfitting, "correctly" fitting and overfitting.

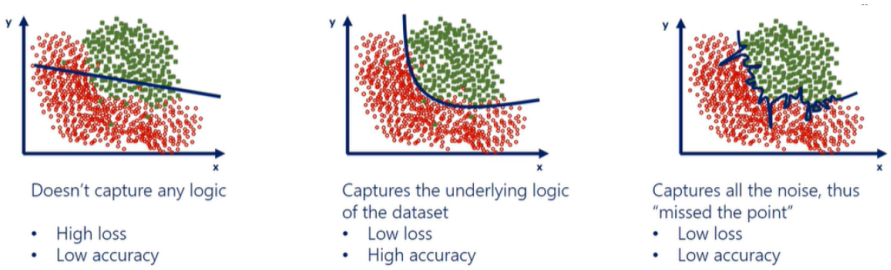

Figure 1: The decision boundary represented by the blue line can be fitted in a vast array of configurations. A linear boundary (left) is not accurate enough and a perfect line (right) will not performe well on unseen data. Illustration from 365datascience.com [3].

The balance between underfitting and overfitting is referred to as the Bias-variance trade-off. A model with high bias (left pane figure 1) will miss the relations between features and the target variable, leading to underfitting. On the other hand a model with high variance will have a low loss on the training set due to "noise modeling", leading to overfitting [4].

The model capacity, that is; the models ability to model complex relations, is in large decided by the practitioner through hyperparameter tuning [1]. Hyperparameters are different between each classifier and will thus be presented in more detail through section 2.3 alongside the model theory.

Apart from employing well tuned classifiers some key concepts to ensure a good generalization performance include; the data standardization, presented in section 2.2 and the choice of evaluation metrics that fits the goal of the classifier and the data, discussed in section 2.6.

2.2 Data preprocessing

As mentioned, the model goal is to have a high generalization performance. To ensure that the model is presented with unseen data the data basis is split into train and test partition. The training partition is used for model hyperparameter tuning and model fitting. The tuning process is commonly performed with cross validation for smaller sample sizes [1]. The validation set is then aggregated from the training set and will contain the data from $\frac{1}{k}$ of the training set where the k signifies the number of k-folds used.

The classifier performance is indeed connected to the input data. Many common classification algorithms is sensitive to magnitude differences in the features. This is commonly solved by applying a standard scaling to each feature post train-test splitting. The standard scaler is defined in equation (2) [1]. Given a feature value x the scaler subtracts the feature mean μ, effectively centering the data around 0, and divides by the feature standard deviation σ. The latter will create a feature distribution with standard deviation of 1, thus removing the differences in feature magnitude.

$$z = \frac{(x - \mu)}{\sigma} \tag{2}$$

2.3 Classification algorithms

2.3.1 Decision trees

Decision trees form the basis for the ensamble method random forest, however it is also a valuable classifier in itself. As for all methods used in this report, decision trees is a supervised learning algorithm that can be use for both classification and regression tasks. In contrast to regression methods like logistic regression, but alike the other

presented methods, decision trees are non-parametric meaning they do not make assumptions concerning the relation between features and target. This makes them highly flexible.

The decision tree structure is shown (as a part of a random forest) in figure 2. The tree consists of nodes of three categories; root node, internal nodes and leaf nodes. Connecting these are branches. When one "growes" a decision tree the gini impuirity g defined in equation (3) is used for decisions of the optimal split.

$$g = \sum_{i=0}^{K} p_i(1 - p_i) \tag{3}$$

Here p_i is the frequency of observations of class i at the node and K is the number of unique labels. The gini impurity is used at the root of the tree and at every internal node,thus each of thee is to be considered a decision point.

In order to grow the three one can use the CART algorithm. The algorithm essentially looks for a single feature k, and a threshold value for k to perform the split at t_k. Using the ginifactor we search for the minimized cost function

$$C(k, t_k) = \frac{n_l eft}{n} g_{left} + \frac{n_{right}}{n} g_{right} \quad , \tag{4}$$

which will be used recursivly for each internal node until a stopping criterion is met or it is only leef nodes left. In this way the CART algorithm will perform feature selection since it is optimizing with the gini impurity. However, using this approach CART is very sensitive to data samples, and a small change can create a larger change in the tree construction [5]. This is referred to as unstable trees.

If stopping criterions is not implemented a descission tree will performe splits until there are only leef nodes left. This makes descission trees prone to overfitting. To combat this there are common pruning techniques that can be deployed, all wchih is categorized in one of two categories; pre-pruning and post-pruning. Pre-pruning involves stopping the growth of the tree early, often using heuristics such as setting a minimum number of training instances for each leaf or limiting the depth of the tree. Post-pruning, on the other hand, involves growing a full tree and then removing nodes and subtrees to reduce complexity and improve generalization. I will only be conducting pre-pruning. Thus I select to tune the max depth of the descision tree using *'max_depth'*: $[1, 2, 3, 4, 5, 10, 15, 20, 50]$ and minimum number of samples required for a node to split using *'min_samples_leaf'*: $[2, 4, 6]$. The tuning is performed on the training set with 3 kfold cross validation.

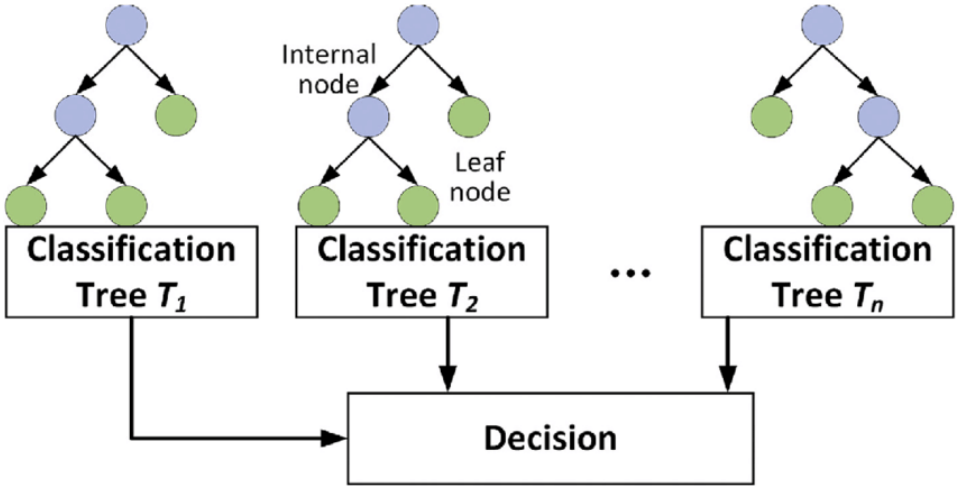

Figure 2: In the random forest classifier an ensamble of decision trees are utalized. The majority vote of the trees is used as the final prediction. Illustration from Zhang et. al (2020 [6]

2.4 Decision trees with bagging: Random forest

As previously mentioned the decision tree algorithm is prone to having high variance. To combat this, random forest uses the bootsrap aggregation (bagging). Figure 2 illustrates how multiple classification trees "vote" for the decisions to be made. This is a key property of the bagging technique. The bagging algorithm performs splits of the training

set with replacement. One decision tree is then trained on the subset, further, introducing the ensamble to the test set the majority vote is used as the classification.

For now we have introduces decision trees with bagging leading to an esamble method. However, we do not want to tune the descission trees in the same way as previously. This will lead to similar trees which in turn removes the purpose of ensambling in the first place [7]. Thus the only hyperparameter of descission trees with bagging is the number of estimators, that is the count of trees in your forest. I will search for this using *'n_estimators'*: $[10, 50, 100, , 500, 1000]$.

To further improve generalization performance I will alter the number of samples that each tree is allowed to check when searching for the optimal split, as given in equation 4. Now we have the the random forest, with a random subset for training through bagging and a reduced number of searches allowed. For the classification task I use the square root of the number of samples defined by *'max_features'*: 'sqrt'.

2.5 Support vector machines

The support vector classifier is another supervised learning algorithm for binary classification. Hwo uses a quite different approach then the tree based methods. The concept of a support vector machine is the fitting of an optimal hyperplane to separate data points with the largest margin possible. Firstly I will present the support vector machine with mathematics before presenting the kernel trick.

Letting one realization being denoted as x_i in the p dimensional space we assume, with a linear kernel, that the binary classes are seperable into two classes $y_i = \pm 1$. For a featurespace of dimension p the separating hyperplane is on of $p-1$ dimensions. Then a linear hyperplane is defined using the general linear model as

$$\mathbf{x} \cdot \tilde{\mathbf{w}} + \tilde{b} = 0 \quad , \tag{5}$$

where $b \in \mathbb{R}$ is the intercept and $\tilde{\mathbf{w}}$ is $\in \mathbb{R}^p$. Considering the two dimensional design matrix for this reports data we have

$$X = [\mathbf{x_{tas}} \quad \mathbf{x_{pr}}] \quad .$$

If the condition of equation (5) is not met we have a classification $y_i = sgn(\mathbf{x} \cdot \tilde{\mathbf{w}} + \tilde{b})1$ as illustrated in figure 3. The placement of the hyperplane is indeed the learning part of the support vector classifier.

For a robust classifier the intuitive approach is placing a hyperplane that maximizes the margin M. Equation (6) presents the constraint for which we which to maximize M.

$$y_i \left(\boldsymbol{x}_i \cdot \tilde{\boldsymbol{w}} + \tilde{b}\right) \geq M \quad \text{or,} \quad y_i \left(\boldsymbol{x_i} \cdot \boldsymbol{w} + b\right) \geq 1 \tag{6}$$

Note that now $||\boldsymbol{w}|| = 1/M$, further for the right hand version we have defined $\boldsymbol{w} := \frac{\tilde{\boldsymbol{w}}}{M}$ and $b = \frac{\tilde{b}}{M}$.

The closes observations relative to the hyperplane are referred to as support vectors, these are represented by the points on the dashed lines in figure 3.

For an easy separable case as figure 3 one could usually perform clustering with a hard-margin SVM, however hard margins does not allow for wrongfull classifications. For more complex relations this could easally lead to overfitting as in the right hand subplot of figure 1 (granted one does not longer use a linear kernel).

For data where noise is more present a soft-margin SVM, that allows for a proportion of the samples (slack) to be on the wrong side of the hyperplane, will be more appropriate [8]. Letting ξ_i be the distance between the slacked samples and the correct class margin we introduce the soft-margin SVM as

$$\begin{aligned} &\min_{\boldsymbol{w},b} \frac{1}{2}\|\boldsymbol{w}\|^2 + C\frac{1}{n}\sum_i \xi_i \\ &\text{subject to } \begin{cases} y_i(\boldsymbol{x} \cdot \boldsymbol{w} + b) \geq (1 - \xi_i) & \text{for } i = 1, \ldots, n \\ \xi_i \geq 0 & \text{for } i = 1, \ldots, n \end{cases} \end{aligned} \tag{7}$$

for some scalar C. C signifies the "cost" of the allowed error by the soft margins. Analyzing equation (7) it is apparent that a smaller C will allow for a greater porportion of points being inside the margins, thus broadening the margin. C

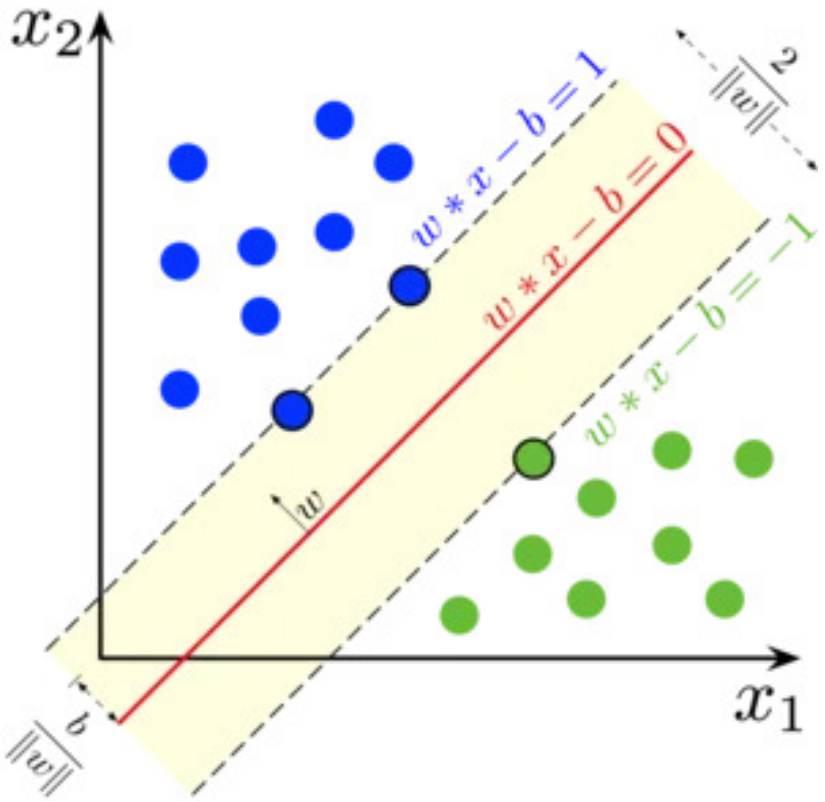

Figure 3: For a two dimensional featurespace the linear kernel is just a linear line function. The line seperates the classes with a margin w and at the edge of the margin we find the support vectors. Illustration from wikipedia.org

is the most important hyperparameter of the support vector classifier, I will be tuning within the range *'C'*: [1.e-03, 1.e-02, 1.e-01, 1.e+00, 1.e+01, 1.e+02] in effort to improve generalization performance.

For non linear hyperplane it is useful to deploy the kernel trick which in essence are transformations of input data into a another (higher-dimensional) space where classification is easier. For this report I have investigated three kernels through *'kernel'*: ['rbf', 'linear', 'sigmoid']. From introducing non-linear kernels a second hyperparameter γ is tuned by *'gamma'*: [1.e-04, 1.e-03, 1.e-02, 1.e-01, 1.e+00]. The γ hyperparameter in the rbf and sigmoid kernel determines the flexibility of the decision boundary and thus, controls its shape.

2.6 Model evaluation

The performance of binary classifications is commonly summarized in a confusion matrix (CM). A generic CM is shown in figure 4 where the correct classifications, True negetive (TN) and True positive (TP) are found on the diagonal. Further we find the False negative (FN) and the False positive (FP) which both are considered an error.

Based on the cells in the confusion matrix we can define multiple metrics regarding the performance of the model. I will briefly introduce four common metrics and their usage.

The *accuracy* is a meassure of the number of correct classifications

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}} = \frac{\text{Correct Predictions}}{\text{Total Predictions}} \quad . \tag{8}$$

Since it uses all cells of the confusion matrix it is a good choice for data sets where the classes are well balanced.

The *precision* is the ratio of predictions set to positive that are actually positiv, that is

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \quad , \tag{9}$$

and so the precission is a powerful metric for data where the FP is of greater importance then the FN.

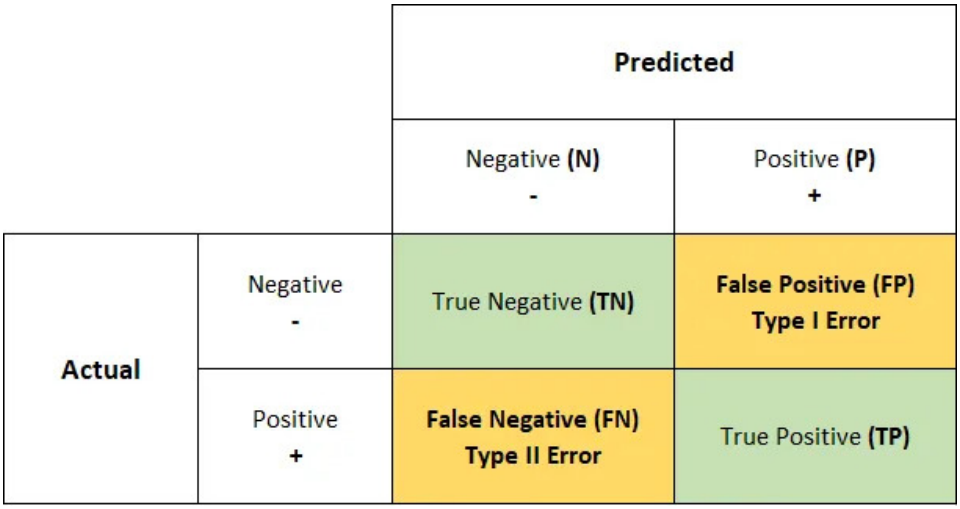

Figure 4: The confusion matrix will effectively summarize the performance of a classification model. Illustration from medium.com/analytics-vidhya [9].

The *recall* is used to measure how many of actual positive samples where classified wrongly as negatives FN

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \quad . \tag{10}$$

The final metric is the *F1-Score* and combines the properties of equation (9) and (10) into equation (11).

$$\text{F1-Score} = 2 * \frac{(\text{ Recall } * \text{ Precision })}{(\text{ Recall } + \text{ Precision })} \tag{11}$$

Combining the properties, F1-score is balancing the weight between classifying wrongfully (FN, FP). It is a common metric for unbalanced classes and thus powerful when a test-train split is performed across classes without a verification of balance [9].

3 Data and and problem setup

3.1 Data

This report presents theory and result interpretations from a mostly data-analytical point of view. I consider the in-detail physical interpretations to be out of the scope of this report. However, a brief survey of data origin and decisions made in the preprocessing is useful for project completeness an reproduce ability of results. Therefor the following subsections will give a brief, informal introduction to the data origin and processing.

3.1.1 Climate model data

The analysis in this report is conducted based on model output data from the Australian Community Climate and Earth System Simulator earth system model 1.5 (ACCESS-ESM1.5). ACCESS-ESM1.5 is a part of the Coupled Model Intercomparison Project Phase 6 (CMIP6), a project founded on the belif that ensambles om models is more representative [10]. To access CMIP6 ACCESS-ESM1.5 has passed "the deck". A set of benchmark simulations ensuring that ACCESS-ESM1.5 is representably tuned for the earth system, the deck is mandatory to pass for the EA model to be allowed to contribute in CMIP6 [11].

The data for this project is generated as a part of simulations of shared socieconomic pathways (SSP's), that is narratives of development for factors that will change climate gas concentrations. The SSP's are closely linked to representative concentration pathways (RCPs) which are direct numbers for change in radiative forcing [Wm^{-2}] in 2100 compared to 1750 levels [12]. In this report we are mainly concerned with two pathways; SSP126 and SSP585, where we have SSP1 and 5 respectively and ΔRF of 2.6 and 8.5Wm^{-2} respectively. For more, and concise, information I suggest visiting this article from climatehubs.usda.gov.

The ACCESS-ESM1.5 model is then run under different SSPs to explore the earth system response to societal choices. Simulations are done historically up to and including 2014 before starting SSPs at 2015. Through the work with this report I have investigated output from SSP126, 245, 370 and 585. I use two features percipitations (pr), and temperature at the surface (tas) to span a featurespace $\in \mathbb{R}^{n \times n}$. This feature space consists of 40 ensemble members (realizations from now) from each SSP. Further processing is outlined in section 3.1.2.

3.1.2 Dimensional reduction

Model output is delivered as a time series with a temporal resolution Δt = 6hours and three spatial coordinates latitude, longitude and vertical position (elevation). From this an annual-temporal and global-spatial climatology is generated yielding one point observation per year for each realization. Figure 5 shows the global annual climatologies from SSP126 ensemble with all realizations.

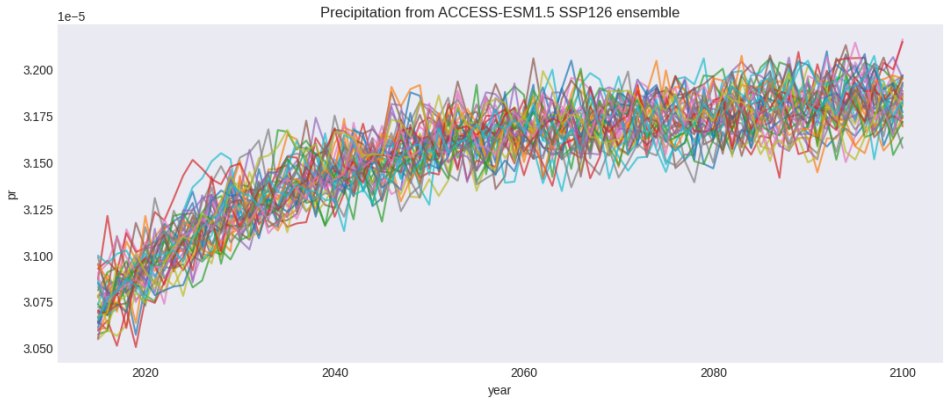

Figure 5: Each realization from ACCESS-ESM1.5 under the SSP126 is represented by one lineplot. Evidently the intra ensamble variability is noisy.

Figure 6 illustrates key features through the ensemble means with variability. There is a clear overlap of each SSP up to about 2040 before they start separating their way. In order to make the classification task easier I will only use SSP126 and SSP585 for further analysis.

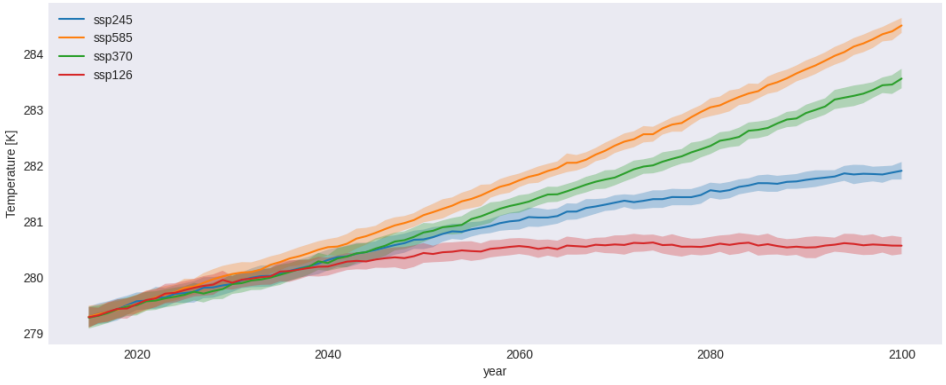

Figure 6: The SSP means with standard deviation for the temperature feature.

The time series begs the question, are there benefits to including history in the classification? For this analysis the history is not considered, rather a a cross sectional approach, as illustrated in **??**, is concluded to be more feasible approach. We therefor leave the temporal approaches for the rest of this report. Referring to the noisy time series shown in figure 5 I will smooth the realizations using a moving window of three years to calculate each cross section. The cross sections will be spanned by the features pr and tas. Further since we are interested in early emerging signals I use the windows $\{2015-2017, 2018-2021, ..., 2039-2041\}$ for the analysis. In agreement with figure 6, figure 7 shows an increasing temperature and precipitation for both SSPs. Further key properties is the gradual separation of pathways as time passes. As is physically expected SSP585, where the ΔRF = 8.5 Wm^{-2} for 2100, is heating faster. There are less separation to be found in the perception feature, where both SSPs show approximately the same variability and mean value.

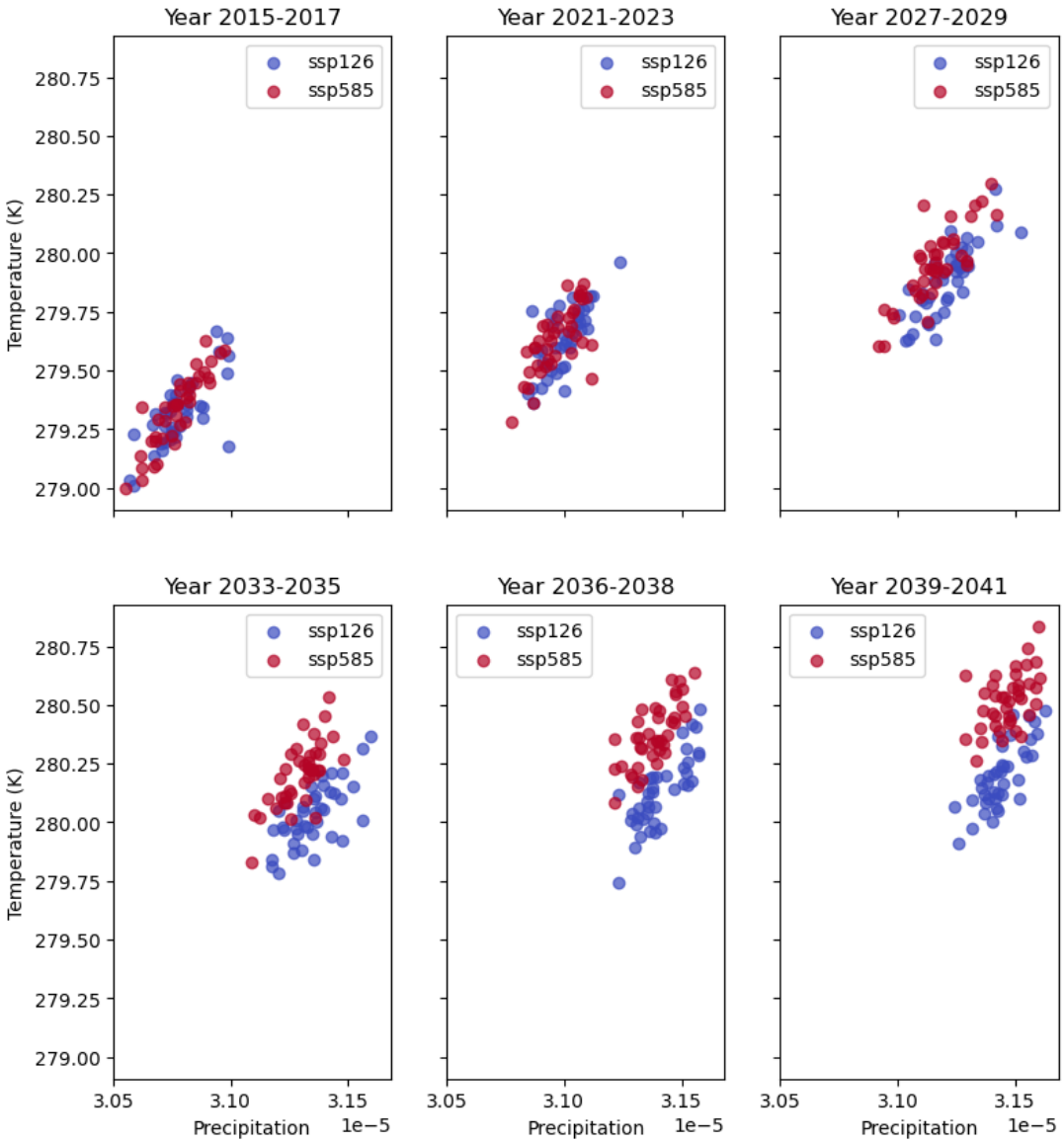

Figure 7: SSP126 and SSP585 starts off without any separation but is gradually separated towards 2040. Note that this is not all periods considered in the full analysis.

4 Results and discussion

The following presentation and discussion of the results will be twofold. The first part will be dedicated to classification performance, purely from a machine learning point of view. Secondly, many simplifications has been done in order to apply the classification algorithms. I will start by discussing the validity of choices, thereunder why the temporal axis is not taken into consideration, feature and data selection.

4.1 Classification performance

The performance of the tuned classification models is displayed in figure 8. From this plot, if one comapares the left plot (seeded 2222) and the right (seeded 99) there is a clear difference in seeding for model performance. This is a response to the small sample size both for training and testing. Further the effect of the seed is worse for the early periods, this is likely due to class overlay as displayed in figure 6 and 7. For a further comparison see the table in appendix B.

Further seed 99 seems to represent a harder split with more test samples in the border region between the classes. This leeds to overfitting as seen by the test scores being much lower then the training. With the threshold for a successful classification at f1 score of 0.8 we see from figure **??** that both the Support vector classifier and the random

forest can be considered sufficient as the test f1 score crosses the 0.8 threshold at 2033-2035 and 2036-2038 period respectively.

In seed 2222 the split is probably with test samples being further away from the class border, since the generalization performance is similar or better then the training score. Here all classifiers cross the 0.8 threshold, respectivly at 2027-2029, 2030-2032 and 2033-2035 for SVC, RF and DT.

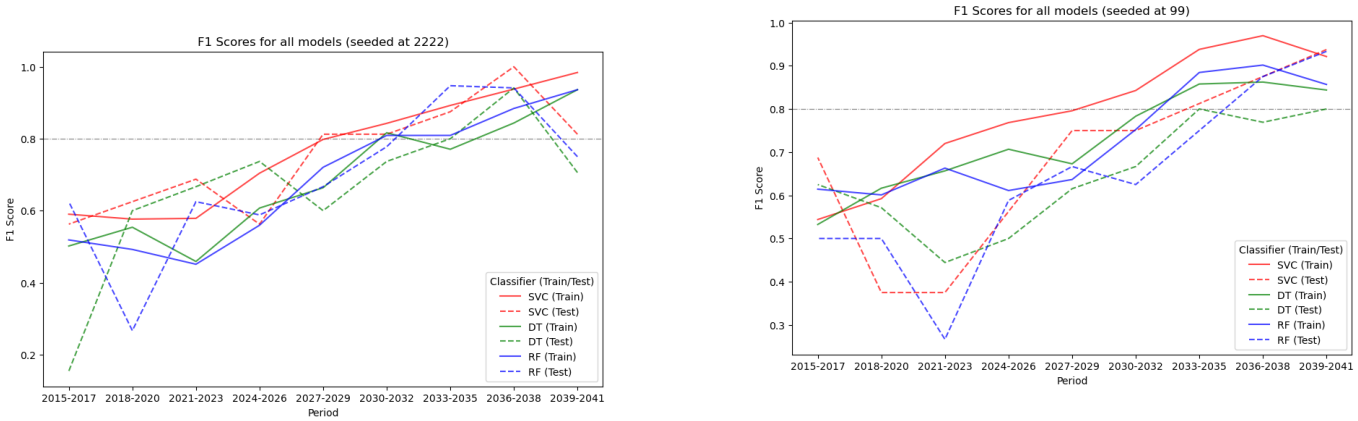

Figure 8: The f1 score development for the periods chosen. From the figures two things are clear. 1. The model performance evaluated by the test f1 score is increasing through the periods. 2. There is a difference between seeds which alters the train-test split.

The f1 score does not indicate where the classification error occurs, this can be investigated using figure 9 where we have the confussion matrices for 2030-2032 and 2036-2038 seeded at 2222. As I did not check the class balance post split I can not assess weather or not one class is harder to classify then the other. However, in our case false positive and false negative classifications will be weighted equally. In an extended analysis consisting of more SSPs the consequences of different classifications will be more significant.

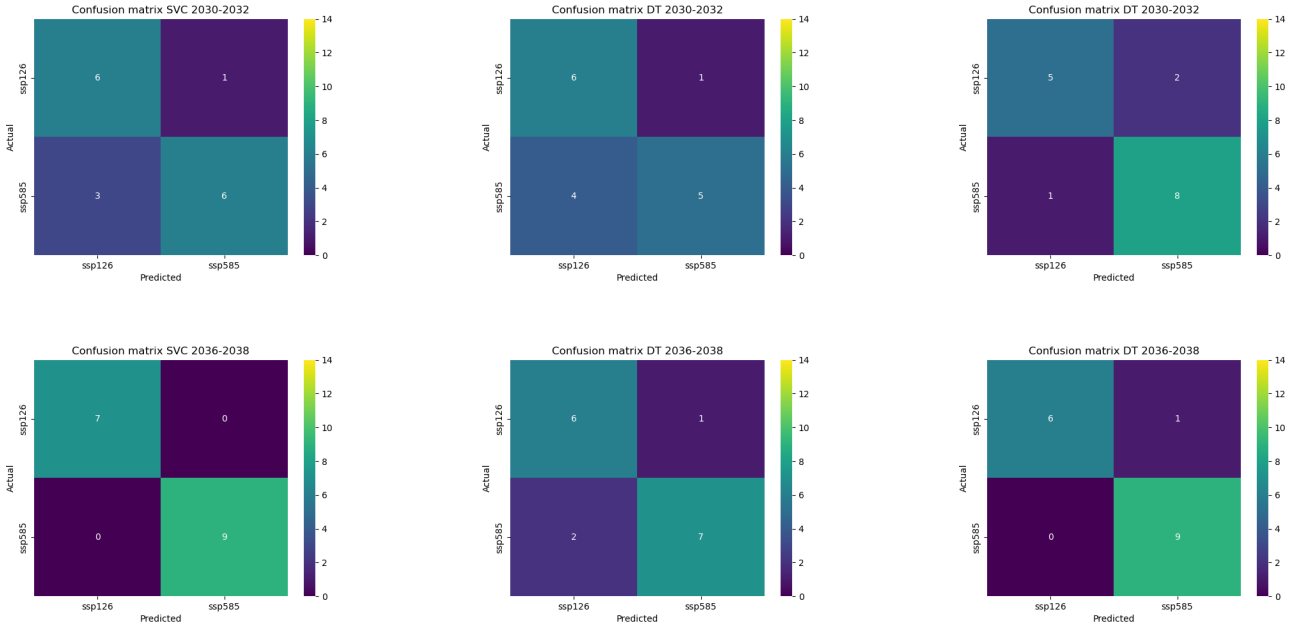

Figure 9: The confusion matrices for two periods of 2030-2032 and 2036-2038. From the left to the right we have the support vector, descission tree and random forest classifier.

To conclude the model performance analysis the support vector classifier has the best over all performance across periods and seeds and would be the prefered classifier. It is however a less flexible classifier for multi modal classification (including more SSPs).

4.2 Validity of assumptions and future work

4.2.1 Temporal dependencies

In order to choose if the temporal relations should be taken into account when classifying one key property for the process is to investigate the autocorrelation $\text{acf}(x_t)$, that is the correlation between the time series and lagged versions. A significant acf for h indicates that a current time point x_t is dependent on previous time steps x_{t-h}. The acf is shown to be significant for multiple ssp which suggest that there is information to be found in the history. Clustering of time series data is typically approached in three different ways:

1. **Distance based classification with hirarchical clustering:** The main idea is to calculate distances between values from different relisations at time t. Figure 6 rules this for early signal detection as the ensemble's are overlapping up to about 2040.
2. **Feature based classificastion:** The main idea is to extract time series features such as wavlet transforms and acf [13]. This not a feasible as the realizations are to noisy.
3. **Model based classification:** The main idea is to train forecasting models and use the model parameters, and hyperparameters for classification. The most promising of well investigated models base on data structure is then ARIMA-forecasting [14]. A typical term that could separate scenarios are differences in referencing degree which is used to make time series stationary with respect to trend. Preliminary investigations shows model training with autoarima yields parameters that are to overlapping. Thus there is no early emerging signal for the model based approach either.

Building upon the discussion above I concluded with that a cross sectional approach is more fesiable. Further investigations of the temporal dependencies should be done. To recap; the data was downsampled from a 6 hour to an anual temporal resolution, effectivly removing all seasonality from the data. This smoothing is excessive if a ts-clustering approach i to be used, on the other hand, some downsampling will be necessary in order to remove sesonalities connected to the 24 h rotational cycle of earth and for noise reduction of the data. Further investigations should include sampling for seasonal periods DJF, MAM, JJA, SON which will open for a wast aray of classification methods. Most intriging would be

a) The creation of a baseline time series per ssp and applying a distance based clustering that is sensitiv to amplitude shifts in order to detect similarity between SSPs and unseen samples.

b) Further distance based clustering should be investigating using the acf distance metric as the acf was shown in my analysis to have large interSSP differences.

c) The model based approach should be investigated again, using SeasonalARIMA model with multiplicative residuals which allows for amplitudal changes of seasonal patterns.

4.2.2 Features and further data selection

In most sciences the system consists of describing variables and response variables. For the climate system typically descriptive variables include the concentrations of trace gases in the atmosphere, solar forcing etc, both of which is measurable. The response variables on the other hand include meteorological factors such as temperature, precipitation, circulation patterns etc which also is measurable in situ. Even both feature categories are measurable in the real world the descibing features are not the main goal to describe, for instance: one are not interested in the CO_2 conentrations because they are harmfull in them selves (broadly speaking, exceptions is known), we are interested in their effect on the climate system, which in turn changes the conditions for life.

Therefore, in this report I use temperature at surface and precipitation, two features known to be intensified by global warming. I chose these as they are;

- Easy to interpret, and
- easy to measure in-situ

compared to other features. The synthesis of report IPCC ar6 concludes with about 20 years in order to seperate SSPs using with high confidence [15]. To take advantage of machine learning algorithms ability to learn complex relations it is probable that an extended analysis including more features would improve model performance.

Further I have only used two SSPs. This remove some of the value of the classifier as it will only represent two quite distinct pathways for the social development. A more well rounded model including more pathways will more efficiently give a feedback of weather or not policies are working.

In this analysis only one model was include, ACCESS-ESM1.5. This is a concern arising as a result of models having different representations of the earth system. Thus the generlization performance may be artificially high since the test data is only a subset of the ACCESS-ESM1.5 output. The ultimate goal of a model like this is classifying in-situ observations to SSP. Following the ensambling principle of the CMIP6 the validity of the classifier will be stronger if it was trained on multiple models.

Lastly the spatial resolution should be further investigated. By aggregating a global mean we effectively perform a spatial signal smoothing. Since the scientific consensus is that arctic and polar regions will experience a more rapid heating, there is information to be found in subset investigations.

Considering the above, there are a wast array of improvements to be done in order to hopefully improve model performance, but more importantly the applicability to real world data with high confidence.

5 Conclusion

In this report, I have examined the intricate relationship between climate policies (represented by climate model data from model runs under different SSPs) using machine learning techniques.

The analysis revealed that the performance of classification models varied significantly depending on the seed used for model performance, highlighting the impact of increasing the sample size when deploying classifiers.

We found that seed 99 seemed to represent a harder split with more test samples in the border region between the classes, leading to overfitting. However, both the Support Vector Classifier and the Random Forest models achieved a test F1 score above the 0.8 threshold, indicating sufficient performance. In contrast, for seed 2222, the split likely resulted in test samples being further away from the class border. All classifiers crossed the 0.8 threshold, and they did so earlier resulting in a better generalization performance compared to seed 99.

Based on the results, the Support Vector Classifier had the best overall performance across periods and seeds and would be the preferred classifier. However, it is less flexible for multi-modal classification, which involves more SSPs.

The results and discussion highlight how this analysis should be extended in multiple directions, most significantly increasing sample size. Further, considering the validity of the assumptions I acknowledged that many simplifications were made to apply the classification algorithms. For future work I suggest a focus on extending sample size, both through increasing realization number but also by including a multi-model approach. Lastly the down sampling should be performed less harshly to allow for further temporal dependency investigation and to catch the dynamics of differential changes across the globe during climate change.

References

[1] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep Learning*. http://www.deeplearningbook.org. MIT Press, 2016.

[2] *Week 41 Neural networks and constructing a neural network code — Applied Data Analysis and Machine Learning*. [Online; accessed 17. Dec. 2023]. Dec. 2023. URL: https://compphysics.github.io/MachineLearning/doc/LectureNotes/_build/html/week41.html?highlight=one%20hot.

[3] *What is Overfitting in Deep Learning [+10 Ways to Avoid It]*. [Online; accessed 17. Dec. 2023]. Dec. 2023. URL: https://www.v7labs.com/blog/overfitting.

[4] *5. Resampling Methods — Applied Data Analysis and Machine Learning*. [Online; accessed 17. Dec. 2023]. Dec. 2023. URL: https://compphysics.github.io/MachineLearning/doc/LectureNotes/_build/html/chapter3.html?highlight=bias.

[5] *9. Decision trees, overarching aims — Applied Data Analysis and Machine Learning*. [Online; accessed 17. Dec. 2023]. Dec. 2023. URL: https://compphysics.github.io/MachineLearning/doc/LectureNotes/_build/html/chapter6.html.

[6] Mengyun Zhang et al. "Fully convolutional networks for blueberry bruising and calyx segmentation using hyperspectral transmittance imaging". In: *Biosyst. Eng.* 192 (Feb. 2020), p. 159. ISSN: 1537-5110. DOI: 10.1016/j.biosystemseng.2020.01.018.

[7] Jason Brownlee. "Bagging and Random Forest Ensemble Algorithms for Machine Learning - MachineLearningMastery.com". In: *MachineLearningMastery* (Dec. 2020). URL: https://machinelearningmastery.com/bagging-and-random-forest-ensemble-algorithms-for-machine-learning.

[8] PhD Zijing Zhu. "Explain Support Vector Machines in Mathematic Details". In: *Medium* (Dec. 2021). URL: https://towardsdatascience.com/explain-support-vector-machines-in-mathematic-details-c7cc1be9f3b9.

[9] Anuganti Suresh. "What is a confusion matrix? - Analytics Vidhya - Medium". In: *Medium* (Dec. 2021). URL: https://medium.com/analytics-vidhya/what-is-a-confusion-matrix-d1c0f8feda5.

[10] *ACCESS-ESM1.5 - Australian Community Climate and Earth System Simulator (ACCESS)*. [Online; accessed 17. Dec. 2023]. Oct. 2021. URL: https://research.csiro.au/access/about/esm1-5.

[11] *GEO4962: The General Circulation of the Atmosphere: Introduction on climate models*. [Online; accessed 17. Dec. 2023]. Aug. 2023. URL: https://metos-uio.github.io/GEO4962/01-Introduction/index.html.

[12] *What are climate model phases and scenarios? | USDA Climate Hubs*. [Online; accessed 17. Dec. 2023]. Dec. 2023. URL: https://www.climatehubs.usda.gov/hubs/northwest/topic/what-are-climate-model-phases-and-scenarios.

[13] Elizabeth Ann Maharaj. *Time Series Clustering and Classification*. CRC Press, Taylor Francis Group, 2019.

[14] Hyndman. R and Athanasopoulos. G. *Forecasting: Principles and Practice (3rd ed)*. Dec. 2023. URL: https://otexts.com/fpp3.

[15] *AR6 Synthesis Report: Climate Change 2023 — IPCC*. [Online; accessed 21. Dec. 2023]. Dec. 2023. URL: https://www.ipcc.ch/report/sixth-assessment-report-cycle.

A Project code

The experiment code is openly available as a GitHub repository:

FYS-STK4155_project_3_copy_sigma2_21_12

The project 3 folder contains two main folders:

- *"scr"*: The src folder mainly contains *.py* files containing classes used for task solving.
- *"conda_envs yaml"*: Contains yaml file for all needed libraries to run code.

Further there is a cronological walkthrough 1-6 for data manipulation and classification. Note that not all code is available to run since you will need access to the main file storage of sigma2.nird.

The two other folders of the repository contains results (figures and some tables) as well as data files with the climatology making the latter codes of 1-6 run-able.

B Table for seed comparison

The following table can be used for seed comparison of different seeds. As the sample size is small there is a large variation in estimated generalization performance signified by the F1 score of test data. This is true for all classifiers.

Period	F1 scores Seed 2022		F1 scores Seed 666		F1 scores Seed 99		Classifier
	Train	Test	Train	Test	Train	Test	
2039-2041	0,984	0,813	0,944	0,800	0,921	0,938	SVC
2036-2038	0,938	1,000	0,972	0,857	0,970	0,875	SVC
2033-2035	0,892	0,875	0,903	0,857	0,938	0,813	SVC
2030-2032	0,843	0,813	0,864	0,769	0,843	0,750	SVC
2027-2029	0,798	0,813	0,846	0,714	0,796	0,750	SVC
2024-2026	0,704	0,563	0,788	0,533	0,768	0,563	SVC
2021-2023	0,579	0,688	0,694	0,545	0,720	0,375	SVC
2018-2020	0,576	0,625	0,694	0,545	0,592	0,375	SVC
2015-2017	0,590	0,563	0,694	0,545	0,544	0,688	SVC
2039-2041	0,936	0,706	0,919	0,750	0,844	0,800	DT
2036-2038	0,844	0,941	0,874	0,857	0,862	0,769	DT
2033-2035	0,771	0,800	0,807	0,857	0,858	0,800	DT
2030-2032	0,817	0,737	0,761	0,714	0,783	0,667	DT
2027-2029	0,664	0,600	0,724	0,462	0,673	0,615	DT
2024-2026	0,607	0,737	0,617	0,182	0,707	0,500	DT
2021-2023	0,459	0,667	0,595	0,429	0,657	0,444	DT
2018-2020	0,554	0,600	0,651	0,471	0,617	0,571	DT
2015-2017	0,502	0,154	0,457	0,444	0,533	0,625	DT
2039-2041	0,936	0,750	0,943	0,800	0,857	0,933	RF
2036-2038	0,884	0,941	0,923	0,857	0,902	0,875	RF
2033-2035	0,809	0,947	0,898	0,800	0,885	0,750	RF
2030-2032	0,809	0,778	0,862	0,706	0,752	0,625	RF
2027-2029	0,721	0,667	0,772	0,364	0,637	0,667	RF
2024-2026	0,559	0,588	0,671	0,400	0,611	0,588	RF
2021-2023	0,451	0,625	0,645	0,400	0,663	0,267	RF
2018-2020	0,492	0,267	0,661	0,533	0,601	0,500	RF
2015-2017	0,519	0,625	0,566	0,667	0,614	0,500	RF