1 Problem statement

The goal is to forecast a dependent variable $\mathbf{y} \in \mathbb{R}^r$ with r targets from an independent input object $\mathbf{x} \in \mathbb{R}^n$ with n features. We assume there is a linear dependence

$$\mathbf{y} = \mathbf{\Theta}\mathbf{x} + \boldsymbol{\varepsilon} \tag{1}$$

between the objects $\mathbf{x}$ and the target variable $\mathbf{y}$, where $\mathbf{\Theta} \in \mathbb{R}^{r \times n}$ is the matrix of model parameters, $\boldsymbol{\varepsilon} \in \mathbb{R}^r$ is the residual vector. The task is to find the matrix of the model parameters $\mathbf{\Theta}$ given a dataset $(\mathbf{X}, \mathbf{Y})$, where $\mathbf{X} \in \mathbb{R}^{m \times n}$ is a design matrix, $\mathbf{Y} \in \mathbb{R}^{m \times r}$ is a target matrix

$$\mathbf{X} = [\mathbf{x}_1, \dots, \mathbf{x}_m]^T = [\boldsymbol{\chi}_1, \dots, \boldsymbol{\chi}_n]; \quad \mathbf{Y} = [\mathbf{y}_1, \dots, \mathbf{y}_m]^T = [\boldsymbol{\nu}_1, \dots, \boldsymbol{\nu}_r].$$

The columns $\boldsymbol{\chi}_j$ of the matrix $\mathbf{X}$ respond to object features. The examples of how to construct the dataset for a particular application task are described in Section Computational experiment.

The optimal parameters are determined by minimization of an error function. Define the quadratic error function:

$$S(\mathbf{\Theta}|\mathbf{X}, \mathbf{Y}) = \Big\| \underset{m \times r}{\mathbf{Y}} - \underset{m \times n}{\mathbf{X}} \cdot \underset{r \times n}{\mathbf{\Theta}}^T \Big\|_2^2 \to \min_{\mathbf{\Theta}}. \tag{2}$$

The solution of the problem (2) is given by

$$\mathbf{\Theta} = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{Y}.$$

The linear dependent columns of the matrix $\mathbf{X}$ leads to an instable solution for the optimization problem (2). If there is a vector $\boldsymbol{\alpha} \neq 0$ such that $\mathbf{X}\boldsymbol{\alpha} = 0$, then adding the vector $\boldsymbol{\alpha}$ to any column of the matrix $\mathbf{\Theta}$ does not change the error function $S(\mathbf{\Theta}|\mathbf{X}, \mathbf{Y})$. In this case the matrix $\mathbf{X}^T\mathbf{X}$ is not invertible. To avoid the strong linear dependence, feature selection and dimensionality reduction techniques are used.

2 Feature selection

The feature selection goal is to find the index set $\mathcal{A} = \{1, \dots, n\}$ of the matrix $\mathbf{X}$ columns. To select the set $\mathcal{A}$ among all possible $2^n - 1$ subsets, introduce the feature selection quality criteria

$$\mathcal{A} = \underset{\mathcal{A}' \subseteq \{1, \dots, n\}}{\arg\max} \, Q(\mathcal{A}'|\mathbf{X}, \mathbf{Y}). \tag{3}$$

Once the solution $\mathcal{A}$ for the problem (3) is known, the problem (2) becomes

$$S(\mathbf{\Theta}_\mathcal{A}|\mathbf{X}_\mathcal{A}, \mathbf{Y}) = \big\| \mathbf{Y} - \mathbf{X}_\mathcal{A}\mathbf{\Theta}_\mathcal{A}^T \big\|_2^2 \to \min_{\mathbf{\Theta}_\mathcal{A}}, \tag{4}$$

where the subscript $\mathcal{A}$ indicates columns with indices from the set $\mathcal{A}$.

2.1 Quadratic Programming Feature Selection

One of the approach to the feature selection is to maximize feature relevances and minimize pairwise feature redundancy. The QPFS algorithm selects non-correlated features, which are relevant to the target vector $\boldsymbol{\nu}$ for the linear regression problem ($r = 1$)

$$\|\boldsymbol{\nu} - \mathbf{X}\boldsymbol{\theta}\|_2^2 \to \min_{\boldsymbol{\theta} \in \mathbb{R}^n}.$$

Introduce two functions: $\text{Sim}(\mathbf{X})$ and $\text{Rel}(\mathbf{X}, \boldsymbol{\nu})$. The $\text{Sim}(\mathbf{X})$ measures the redundancy between features, the $\text{Rel}(\mathbf{X}, \boldsymbol{\nu})$ contains relevances between each feature and the target vector $\boldsymbol{\nu}$. We want to minimize the function Sim and maximize the Rel simultaneously.

QPFS offers the explicit way to construct the functions Sim and Rel. The method minimizes the following functional

$$(1-\alpha) \cdot \underbrace{\mathbf{a}^T\mathbf{Q}\mathbf{a}}_{\text{Sim}} - \alpha \cdot \underbrace{\mathbf{b}^T\mathbf{a}}_{\text{Rel}} \to \min_{\substack{\mathbf{a} \geq \mathbf{0}_n \\ \mathbf{1}_n^T\mathbf{a}=1}}. \tag{5}$$

The matrix $\mathbf{Q} \in \mathbb{R}^{n \times n}$ entries measure the pairwise similarities between features. The vector $\mathbf{b} \in \mathbb{R}^n$ expresses the similarities between each feature and the target matrix $\mathbf{b}$. The normalized vector $\mathbf{a}$ shows the importance of each feature. The functional (5) penalizes the dependent features by the function Sim and encourages features relevant to the target by the function Rel. The parameter α allows to control the trade-off between the functions Sim and the Rel. The authors of the original QPFS paper suggested the way to select α and make $\text{Sim}(\mathbf{X})$ and $\text{Rel}(\mathbf{X}, \boldsymbol{\nu})$ impact the same

$$\alpha = \frac{\overline{\mathbf{Q}}}{\overline{\mathbf{Q}} + \overline{\mathbf{b}}},$$

where $\overline{\mathbf{Q}}$, $\overline{\mathbf{b}}$ are the mean values of $\mathbf{Q}$ and $\mathbf{b}$ respectively. Apply the thresholding for $\mathbf{a}$ to find the optimal feature subset:

$$j \in \mathcal{A} \Leftrightarrow a_j > \tau.$$

To measure similarity the authors use the absolute value of sample correlation coefficient between pairs of features for the function Sim, and between features and the target vector $\boldsymbol{\nu}$ for the function Rel

$$\mathbf{Q} = \big\{|\text{corr}(\boldsymbol{\chi}_i, \boldsymbol{\chi}_j)|\big\}_{i,j=1}^n, \quad \mathbf{b} = \{|\text{corr}(\boldsymbol{\chi}_i, \boldsymbol{\nu})|\}_{i=1}^n. \tag{6}$$

The problem (5) is convex if the matrix $\mathbf{Q}$ is positive semidefinite. In general it is not always true. To satisfy this condition, the matrix $\mathbf{Q}$ spectrum is shifted and the matrix $\mathbf{Q}$ is replaced by $\mathbf{Q} - \lambda_{\min}\mathbf{I}$, where $\lambda_{\min}$ is a $\mathbf{Q}$ minimal eigenvalue.

The functional (5) corresponds to the quality criteria $Q(\mathcal{A}|\mathbf{X}, \boldsymbol{\nu})$

$$\mathcal{A} = \underset{\mathcal{A}' \subseteq \{1,\dots,n\}}{\arg\max}\, Q(\mathcal{A}'|\mathbf{X}, \boldsymbol{\nu}) \Leftrightarrow \underset{\mathbf{a} \geq \mathbf{0}_n,\, \mathbf{1}_n^T\mathbf{a}=1}{\arg\min} \big[\mathbf{a}^T\mathbf{Q}\mathbf{a} - \alpha \cdot \mathbf{b}^T\mathbf{a}\big]. \tag{7}$$

2.2 Multivariate QPFS

Relevance aggregation First approach to apply the QPFS algorithm to the multivariate case ($r > 1$) is to aggregate feature relevances through all r components. The term $\text{Sim}(\mathbf{X})$ is still the same, and the matrix $\mathbf{Q}$ and the vector $\mathbf{b}$ are equal to

$$\mathbf{Q} = \left\{ \left| \text{corr}(\boldsymbol{\chi}_i, \boldsymbol{\chi}_j) \right| \right\}_{i,j=1}^{n}, \quad \mathbf{b} = \left\{ \sum_{k=1}^{r} |\text{corr}(\boldsymbol{\chi}_i, \boldsymbol{\nu}_k)| \right\}_{i=1}^{n}.$$

This approach does not use the dependencies in the columns of the matrix $\mathbf{Y}$. Let consider the following example:

$$\mathbf{X} = [\boldsymbol{\chi}_1, \boldsymbol{\chi}_2, \boldsymbol{\chi}_3], \quad \mathbf{Y} = [\underbrace{\boldsymbol{\nu}_1, \boldsymbol{\nu}_1, \dots, \boldsymbol{\nu}_1}_{r-1}, \boldsymbol{\nu}_2],$$

We have three features and r targets, where first $r-1$ target are the identical. The pairwise features similarities are given by the matrix $\mathbf{Q}$. Matrix $\mathbf{B}$ entries shows pairwise relevances features to the targets. The vector $\mathbf{b}$ is obtained by summation of the matrix $\mathbf{B}$ over columns.

$$\mathbf{Q} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0.8 \\ 0 & 0.8 & 1 \end{bmatrix}, \quad \mathbf{B} = \left[\begin{matrix} \underbrace{\begin{matrix} 0.4 & \dots & 0.4 \\ 0.5 & \dots & 0.5 \\ 0.8 & \dots & 0.8 \end{matrix}}_{r-1} & \begin{matrix} 0 \\ 0.8 \\ 0.1 \end{matrix} \end{matrix} \right], \quad \mathbf{b} = \begin{bmatrix} (r-1) \cdot 0.4 + 0 \\ (r-1) \cdot 0.5 + 0.8 \\ (r-1) \cdot 0.8 + 0.1 \end{bmatrix}$$

We would like to select only two features. For such configuration the best feature subset is $[\boldsymbol{\chi}_1, \boldsymbol{\chi}_2]$. The feature $\boldsymbol{\chi}_2$ predicts the second target $\boldsymbol{\nu}_2$ and the combination of features $\boldsymbol{\chi}_1, \boldsymbol{\chi}_2$ predicts the first component. The QPFS algorithm for $r = 2$ gives the solution $\mathbf{a} = [0.37, 0.61, 0.02]$. It coincides with our knowledge. However, if we add the collinear columns to the matrix $\mathbf{Y}$ and increase r to 5, the QPFS solution will be $\mathbf{a} = [0.40, 0.17, 0.43]$. Here we lost the relevant feature $\boldsymbol{\chi}_2$ and select the redundant feature $\boldsymbol{\chi}_3$.

Symmetric importances To take into account the dependencies in the columns of the matrix $\mathbf{Y}$ we extend the QPFS functional (5) to the multivariate case. We add the term $\text{Sim}(\mathbf{Y})$ and extend the term $\text{Rel}(\mathbf{X}, \mathbf{Y})$:

$$\alpha_1 \cdot \underbrace{\mathbf{a}_x^{T} \mathbf{Q}_x \mathbf{a}_x}_{\text{Sim}(\mathbf{X})} - \alpha_2 \cdot \underbrace{\mathbf{a}_x^{T} \mathbf{B} \mathbf{a}_y}_{\text{Rel}(\mathbf{X},\mathbf{Y})} + \alpha_3 \cdot \underbrace{\mathbf{a}_y^{T} \mathbf{Q}_y \mathbf{a}_y}_{\text{Sim}(\mathbf{Y})} \to \min_{\substack{\mathbf{a}_x \geq \mathbf{0}_n, \, \mathbf{1}_n^{T} \mathbf{a}_x = 1 \\ \mathbf{a}_y \geq \mathbf{0}_r, \, \mathbf{1}_r^{T} \mathbf{a}_y = 1}}. \tag{8}$$

Determine the entries of matrices $\mathbf{Q}_x \in \mathbb{R}^{n \times n}$, $\mathbf{Q}_y \in \mathbb{R}^{r \times r}$, $\mathbf{B} \in \mathbb{R}^{n \times r}$ in the following way

$$\mathbf{Q}_x = \left\{ \left| \text{corr}(\boldsymbol{\chi}_i, \boldsymbol{\chi}_j) \right| \right\}_{i,j=1}^{n}, \quad \mathbf{Q}_y = \left\{ |\text{corr}(\boldsymbol{\nu}_i, \boldsymbol{\nu}_j)| \right\}_{i,j=1}^{r}, \quad \mathbf{B} = \left\{ |\text{corr}(\boldsymbol{\chi}_i, \boldsymbol{\nu}_j)| \right\}_{\substack{i=1,\dots,n \\ j=1,\dots,r}}.$$

The vector $\mathbf{a}_x$ shows the feature importances, while $\mathbf{a}_y$ is a vector with the importance of each target. The targets which are correlated will be penalized by $\text{Sim}(\mathbf{Y})$ and have the lower importances.

The coefficients α_1, α_2, and α_3 control the influence of each term to the functional (8) and satisfy the conditions:

$$\alpha_1 + \alpha_2 + \alpha_3 = 1 \quad \alpha_i \geq 0,\, i = 1, 2, 3.$$

Statement 1. *Balance between the terms Sim*$(\mathbf{X})$*, Rel*$(\mathbf{X}, \mathbf{Y})$*, and Rel*$(\mathbf{X}, \mathbf{Y})$ *for the problem* (8) *is achieved by the following coefficients:*

$$\alpha_1 = \frac{\overline{\mathbf{Q}}_y \overline{\mathbf{B}}}{\overline{\mathbf{Q}}_y \overline{\mathbf{B}} + \overline{\mathbf{Q}}_x \overline{\mathbf{Q}}_y + \overline{\mathbf{Q}}_x \overline{\mathbf{B}}}; \quad \alpha_2 = \frac{\overline{\mathbf{Q}}_x \overline{\mathbf{Q}}_y}{\overline{\mathbf{Q}}_y \overline{\mathbf{B}} + \overline{\mathbf{Q}}_x \overline{\mathbf{Q}}_y + \overline{\mathbf{Q}}_x \overline{\mathbf{B}}}; \quad \alpha_3 = \frac{\overline{\mathbf{Q}}_x \overline{\mathbf{B}}}{\overline{\mathbf{Q}}_y \overline{\mathbf{B}} + \overline{\mathbf{Q}}_x \overline{\mathbf{Q}}_y + \overline{\mathbf{Q}}_x \overline{\mathbf{B}}},$$

where $\overline{\mathbf{Q}}_x$*,* $\overline{\mathbf{B}}$*,* $\overline{\mathbf{Q}}_y$ *are the mean values of* $\mathbf{Q}_x$*,* $\mathbf{B}$*, and* $\mathbf{Q}_y$ *respectively.*

Proof. The desired values of α_1, α_2, and α_3 are given by solution of the following equations

$$\begin{aligned} &\alpha_1 + \alpha_2 + \alpha_3 = 1; \\ &\alpha_1 \overline{\mathbf{Q}}_x = \alpha_2 \overline{\mathbf{B}} = \alpha_3 \overline{\mathbf{Q}}_y. \end{aligned}$$

Here, the mean values $\overline{\mathbf{Q}}_x$, $\overline{\mathbf{B}}$, $\overline{\mathbf{Q}}_y$ of the corresponding matrices $\mathbf{Q}_x$, $\mathbf{B}$, and $\mathbf{Q}_y$ are the mean values of the terms $\text{Sim}(\mathbf{X})$, $\text{Rel}(\mathbf{X}, \mathbf{Y})$, and $\text{Rel}(\mathbf{X}, \mathbf{Y})$. □

To investigate the impact of the term $\text{Sim}(\mathbf{Y})$ on the functional (8), we balance the terms $\text{Sim}(\mathbf{X})$ and $\text{Rel}(\mathbf{X}, \mathbf{Y})$ by fixing the proportion between α_1 and α_2:

$$\alpha_1 = \frac{(1 - \alpha_3)\overline{\mathbf{B}}}{\overline{\mathbf{Q}}_x + \overline{\mathbf{B}}}; \quad \alpha_2 = \frac{(1 - \alpha_3)\overline{\mathbf{Q}}_x}{\overline{\mathbf{Q}}_x + \overline{\mathbf{B}}}; \quad \alpha_3 \in [0, 1]. \tag{9}$$

We apply the proposed algorithm to the discussed example. The given matrix $\mathbf{Q}$ corresponds to the matrix $\mathbf{Q}_x$. We additionally define the matrix $\mathbf{Q}_y$ by setting $\text{corr}(\boldsymbol{\nu}_1, \boldsymbol{\nu}_2) = 0.2$ and all others entries to one. Figure 1 shows the importances of features $\mathbf{a}_x$ and targets $\mathbf{a}_y$ with respect to α_3 coefficient. If α_3 is small, the impact of all targets are almost equal and the feature $\boldsymbol{\chi}_3$ dominates the feature $\boldsymbol{\chi}_2$. When α_3 becomes larger than 0.2, the importance $(\mathbf{a}_y)_5$ of the target ϕ_5 grows up along with the importance of the feature $\boldsymbol{\chi}_2$.

Minimax QPFS The functional (8) is symmetric with respect to $\mathbf{a}_x$ and $\mathbf{a}_y$. It penalizes features that are correlated and do not relevant to targets. At the same time it penalizes targets that are correlated and are not sufficiently explained by the features. It leads to small importances for targets which are difficult to predict by features and large importances for targets which are strongly correlated with features. It contradicts with the intuition. Our goal is to predict all targets, especially which are difficult to explain, by selected relevant and non-correlated features. We express this into two related problems.

$$\alpha_1 \cdot \underbrace{\mathbf{a}_x^{T} \mathbf{Q}_x \mathbf{a}_x}_{\text{Sim}(\mathbf{X})} - \alpha_2 \cdot \underbrace{\mathbf{a}_x^{T} \mathbf{B} \mathbf{a}_y}_{\text{Rel}(\mathbf{X}, \mathbf{Y})} \to \min_{\substack{\mathbf{a}_x \geq \mathbf{0}_n, \\ \mathbf{1}_n^{T} \mathbf{a}_x = 1}}; \tag{10}$$

$$\alpha_3 \cdot \underbrace{\mathbf{a}_y^{T} \mathbf{Q}_y \mathbf{a}_y}_{\text{Sim}(\mathbf{Y})} + \alpha_2 \cdot \underbrace{\mathbf{a}_x^{T} \mathbf{B} \mathbf{a}_y}_{\text{Rel}(\mathbf{X}, \mathbf{Y})} \to \min_{\substack{\mathbf{a}_y \geq \mathbf{0}_r, \\ \mathbf{1}_r^{T} \mathbf{a}_y = 1}}. \tag{11}$$

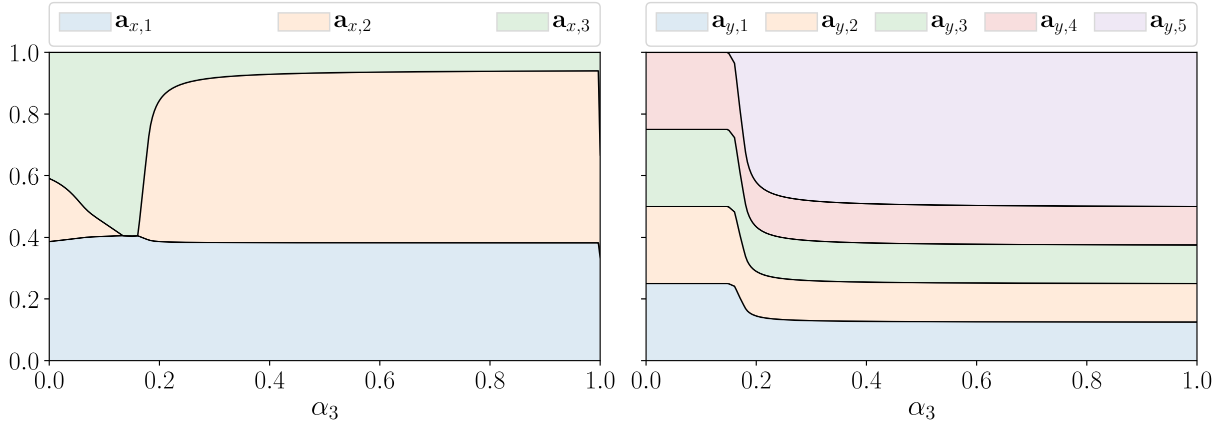


Figure 1: Feature importances $\mathbf{a}_x$ and $\mathbf{a}_y$ with respect to the α_3 coefficient

The difference in Rel part. In feature space the non-relevant components should have smaller scores. Meanwhile, the targets that are not relevant to the features should have larger scores. The problems (10) and (11) are merged into the joint min-max or max-min formulation

$$\min_{\substack{\mathbf{a}_x \geq \mathbf{0}_n \\ \mathbf{1}_n^T \mathbf{a}_x = 1}} \max_{\substack{\mathbf{a}_y \geq \mathbf{0}_r \\ \mathbf{1}_r^T \mathbf{a}_y = 1}} f(\mathbf{a}_x, \mathbf{a}_y), \quad \left(\text{or} \max_{\substack{\mathbf{a}_y \geq \mathbf{0}_r \\ \mathbf{1}_r^T \mathbf{a}_y = 1}} \min_{\substack{\mathbf{a}_x \geq \mathbf{0}_n \\ \mathbf{1}_n^T \mathbf{a}_x = 1}} f(\mathbf{a}_x, \mathbf{a}_y) \right), \tag{12}$$

where

$$f(\mathbf{a}_x, \mathbf{a}_y) = \alpha_1 \cdot \underbrace{\mathbf{a}_x^T \mathbf{Q}_x \mathbf{a}_x}_{\mathrm{Sim}(\mathbf{X})} - \alpha_2 \cdot \underbrace{\mathbf{a}_x^T \mathbf{B} \mathbf{a}_y}_{\mathrm{Rel}(\mathbf{X}, \mathbf{Y})} - \alpha_3 \cdot \underbrace{\mathbf{a}_y^T \mathbf{Q}_y \mathbf{a}_y}_{\mathrm{Sim}(\mathbf{Y})}.$$

The link between feature selection quality criteria (3) and the min-max problem (12) is the following

$$\mathcal{A} = \operatorname*{arg\,max}_{\mathcal{A}' \subseteq \{1, \dots, n\}} Q(\mathcal{A}' | \mathbf{X}, \mathbf{Y}) \Leftrightarrow \operatorname*{arg\,min}_{\mathbf{a}_x \geq \mathbf{0}_n, \mathbf{1}_n^T \mathbf{a}_x = 1} \left[\max_{\substack{\mathbf{a}_y \geq \mathbf{0}_r \\ \mathbf{1}_r^T \mathbf{a}_y = 1}} f(\mathbf{a}_x, \mathbf{a}_y) \right]. \tag{13}$$

Theorem 1. *For positive definite matrices $\mathbf{Q}_x$ and $\mathbf{Q}_y$ the max-min and min-max problems* (12) *have the same optimal value.*

Proof. Denote

$$\mathbb{C}^n = \{\mathbf{a} : \mathbf{a} \geq \mathbf{0}_n, \, \mathbf{1}_n^T \mathbf{a} = 1\}, \quad \mathbb{C}^r = \{\mathbf{a} : \mathbf{a} \geq \mathbf{0}_r, \, \mathbf{1}_r^T \mathbf{a} = 1\};$$

The sets $\mathbb{C}^n$ and $\mathbb{C}^r$ are compact and convex. The function $f : \mathbb{C}^n \times \mathbb{C}^r \to \mathbb{R}$ is a continuous function. If $\mathbf{Q}_x$ and $\mathbf{Q}_y$ are positive definite matrices, the function f is convex-concave, i.e. $f(\cdot, \mathbf{a}_y) : \mathbb{C}^n \to \mathbb{R}$ is convex for fixed $\mathbf{a}_y$, and $f(\mathbf{a}_x, \cdot) : \mathbb{C}^r \to \mathbb{R}$ is concave for fixed $\mathbf{a}_x$. In this case Neumann's minimax theorem states

$$\min_{\mathbf{a}_x \in \mathbb{C}^n} \max_{\mathbf{a}_y \in \mathbb{C}^r} f(\mathbf{a}_x, \mathbf{a}_y) = \max_{\mathbf{a}_y \in \mathbb{C}^r} \min_{\mathbf{a}_x \in \mathbb{C}^n} f(\mathbf{a}_x, \mathbf{a}_y).$$

□

To solve the min-max problem (12), fix some $\mathbf{a}_x \in \mathbb{C}^n$. For fixed vector $\mathbf{a}_x$ we solve the problem

$$\max_{\mathbf{a}_y \in \mathbb{C}_r} f(\mathbf{a}_x, \mathbf{a}_y) = \max_{\substack{\mathbf{a}_y \geq \mathbf{0}_r \\ \mathbf{1}_r^T \mathbf{a}_y = 1}} \big[\alpha_1 \cdot \mathbf{a}_x^T \mathbf{Q}_x \mathbf{a}_x - \alpha_2 \cdot \mathbf{a}_x^T \mathbf{B} \mathbf{a}_y - \alpha_3 \cdot \mathbf{a}_y^T \mathbf{Q}_y \mathbf{a}_y\big]. \tag{14}$$

The Lagrangian for this problem is

$$L(\mathbf{a}_x, \mathbf{a}_y, \lambda, \boldsymbol{\mu}) = \alpha_1 \cdot \mathbf{a}_x^T \mathbf{Q}_x \mathbf{a}_x - \alpha_2 \cdot \mathbf{a}_x^T \mathbf{B} \mathbf{a}_y - \alpha_3 \cdot \mathbf{a}_y^T \mathbf{Q}_y \mathbf{a}_y + \lambda \cdot (\mathbf{1}_r^T \mathbf{a}_y - 1) + \boldsymbol{\mu}^T \mathbf{a}_y.$$

Here the Lagrange multipliers $\boldsymbol{\mu}$, corresponding to the inequality constraints $\mathbf{a}_y \geq \mathbf{0}_r$, are restricted to be nonnegative. The dual problem is

$$\min_{\lambda,\, \boldsymbol{\mu} \geq \mathbf{0}_r} g(\mathbf{a}_x, \lambda, \boldsymbol{\mu}) = \min_{\lambda,\, \boldsymbol{\mu} \geq \mathbf{0}_r} \left[\max_{\mathbf{a}_y \in \mathbb{R}^r} L(\mathbf{a}_x, \mathbf{a}_y, \lambda, \boldsymbol{\mu})\right]. \tag{15}$$

The strong duality holds for the problem (14). Therefore, the optimal value for (14) equals the optimal value for (15). It allows to solve the problem

$$\min_{\mathbf{a}_x \in \mathbb{C}^n,\, \lambda,\, \boldsymbol{\mu} \geq \mathbf{0}_r} g(\mathbf{a}_y, \lambda, \boldsymbol{\mu}) \tag{16}$$

instead of (12).

Setting the gradient of the Langrangian $\nabla_{\mathbf{a}_y} L(\mathbf{a}_x, \mathbf{a}_y, \lambda, \boldsymbol{\mu})$ to zero, we obtain an optimal value $\mathbf{a}_y$:

$$\mathbf{a}_y = \frac{1}{2\alpha_3} \mathbf{Q}_y^{-1} \left(-\alpha_2 \cdot \mathbf{B}^T \mathbf{a}_x + \lambda \cdot \mathbf{1}_r + \boldsymbol{\mu}\right). \tag{17}$$

The dual function is equal to

$$\begin{aligned} g(\mathbf{a}_x, \lambda, \boldsymbol{\mu}) = \max_{\mathbf{a}_y \in \mathbb{R}^r} L(\mathbf{a}_x, \mathbf{a}_y, \lambda, \boldsymbol{\mu}) = \mathbf{a}_x^T \left(-\frac{\alpha_2^2}{4\alpha_3} \mathbf{B} \mathbf{Q}_y^{-1} \mathbf{B}^T - \alpha_1 \cdot \mathbf{Q}_x\right) \mathbf{a}_x \\ - \frac{1}{4\alpha_3} \lambda^2 \cdot \mathbf{1}_r^T \mathbf{Q}_y^{-1} \mathbf{1}_r - \frac{1}{4\alpha_3} \boldsymbol{\mu}^T \mathbf{Q}_y^{-1} \boldsymbol{\mu} + \frac{\alpha_2}{2\alpha_3} \lambda \cdot \mathbf{1}_r^T \mathbf{Q}_y^{-1} \mathbf{B}^T \mathbf{a}_x \\ - \frac{1}{2\alpha_3} \lambda \cdot \mathbf{1}_r^T \mathbf{Q}_y^{-1} \boldsymbol{\mu} + \frac{\alpha_2}{2\alpha_3} \boldsymbol{\mu}^T \mathbf{Q}_y^{-1} \mathbf{B}^T \mathbf{a}_x + \lambda. \end{aligned} \tag{18}$$

It brings to the quadratic problem (16) with $n + r + 1$ variables.

Minimax Relevances The problem (16) is not convex. If we shift the spectrum for the matrix of quadratic form (18), the optimality is lost. To overcome this problem, we drop the term $\text{Sim}(\mathbf{Y})$.

$$\min_{\substack{\mathbf{a}_x \geq \mathbf{0}_n \\ \mathbf{1}_n^T \mathbf{a}_x = 1}} \max_{\substack{\mathbf{a}_y \geq \mathbf{0}_r \\ \mathbf{1}_r^T \mathbf{a}_y = 1}} \left[(1 - \alpha) \cdot \mathbf{a}_x^T \mathbf{Q}_x \mathbf{a}_x - \alpha \cdot \mathbf{a}_x^T \mathbf{B} \mathbf{a}_y\right], \tag{19}$$

The Lagrangian for the problem (19) with the fixed vector $\mathbf{a}_x$ is

$$L(\mathbf{a}_x, \mathbf{a}_y, \lambda, \boldsymbol{\mu}) = (1-\alpha) \cdot \mathbf{a}_x^T \mathbf{Q}_x \mathbf{a}_x - \alpha \cdot \mathbf{a}_x^T \mathbf{B} \mathbf{a}_y + \lambda \cdot (\mathbf{1}_r^T \mathbf{a}_y - 1) + \boldsymbol{\mu}^T \mathbf{a}_y.$$

Setting the gradient of the Langrangian $\nabla_{\mathbf{a}_y} L(\mathbf{a}_x, \mathbf{a}_y, \lambda, \boldsymbol{\mu})$ to zero, we obtain:

$$\alpha \cdot \mathbf{B}^T \mathbf{a}_x = \lambda \cdot \mathbf{1}_r + \boldsymbol{\mu}.$$

The dual function is equal to

$$g(\mathbf{a}_x, \lambda, \boldsymbol{\mu}) = \begin{cases} (1-\alpha) \cdot \mathbf{a}_x^T \mathbf{Q}_x \mathbf{a}_x - \lambda, & \alpha \cdot \mathbf{B}^T \mathbf{a}_x = \lambda \cdot \mathbf{1}_r + \boldsymbol{\mu}; \\ +\infty, & \text{otherwise.} \end{cases} \quad (20)$$

In this case the feature scores is the solution of (16).

Statement 2. *For the case $r = 1$ the proposed functionals* (8) *and* (12) *coincide with the original QPFS algorithm* (5).

Proof. If r is equal to 1, then $\mathbf{Q}_y = q_y$ is a scalar, $\mathbf{a}_y = 1$, $\mathbf{B} = \mathbf{b}$. It reduces the problems (8) and (12) to

$$\alpha_1 \cdot \mathbf{a}_x^T \mathbf{Q}_x \mathbf{a}_x - \alpha_2 \cdot \mathbf{a}_x^T \mathbf{b} \to \min_{\mathbf{a}_x \geq \mathbf{0}_n, \, \mathbf{1}_n^T \mathbf{a}_x = 1}.$$

Setting $\alpha = \frac{\alpha_2}{\alpha_1 + \alpha_2}$ brings to the original QPFS problem (5). □

3 Feature categorization

Feature selection algorithms eliminate features which are not relevant to the target variable. To determine whether the feature is relevant the t-test could be applied for the correlation coefficient.

$$r = \text{corr}(\boldsymbol{\chi}, \boldsymbol{\nu}), \quad t = \frac{r\sqrt{m-2}}{1-r^2} \sim \text{St}(m-2).$$

$$H_0 : r = 0$$
$$H_1 : r \neq 0$$

If features are relevant, but correlated, feature selection methods pick the subset of them to reduce the multicollinearity and redundancy. The goal is to find relevant, non-correlated features. However, in this case the correlations between targets in matrix $\mathbf{Y}$ are crucial. To measure the dependence of each feature or target, the Variance Inflation Factor (VIF) is computed

$$\text{VIF}(\boldsymbol{\chi}_j) = \frac{1}{1-R_j^2}, \quad \text{VIF}(\boldsymbol{\nu}_k) = \frac{1}{1-R_k^2},$$

where $R_j^2(R_k^2)$ are coefficients of determination for the regression of $\boldsymbol{\chi}_j(\boldsymbol{\nu}_k)$ on the other features(targets).

On that basis, we categorize features into 5 disjoint groups:

1. non-relevant features

$$\bigl\{j : \text{corr}(\boldsymbol{\chi}_j, \boldsymbol{\nu}_k) = 0,\ \forall k \in \{1, \dots, r\}\bigr\}\,;$$

2. non-**X**-correlated features, which are relevant to non-**Y**-correlated targets

$$\bigl\{j : \bigl(\text{VIF}(\boldsymbol{\chi}_j) < 10\bigr) \text{ and } \bigl(\text{VIF}(\boldsymbol{\nu}_k) < 10,\ \forall k \in \{1, \dots, r\} : \text{corr}(\boldsymbol{\chi}_j, \boldsymbol{\nu}_k) \neq 0\bigr)\bigr\}\,;$$

3. non-**X**-correlated features, which are relevant to **Y**-correlated targets

$$\bigl\{j : \bigl(\text{VIF}(\boldsymbol{\chi}_j) < 10\bigr) \text{ and } \bigl(\exists k \in \{1, \dots, r\} : \text{VIF}(\boldsymbol{\nu}_k) > 10 \ \&\ \text{corr}(\boldsymbol{\chi}_j, \boldsymbol{\nu}_k) \neq 0\bigr)\bigr\}\,;$$

4. **X**-correlated features, which are relevant to non-**Y**-correlated targets

$$\bigl\{j : \bigl(\text{VIF}(\boldsymbol{\chi}_j) > 10\bigr) \text{ and } \bigl(\text{VIF}(\boldsymbol{\nu}_k) < 10,\ \forall k \in \{1, \dots, r\} : \text{corr}(\boldsymbol{\chi}_j, \boldsymbol{\nu}_k) \neq 0\bigr)\bigr\}\,;$$

5. **X**-correlated features, which are relevant to **Y**-correlated targets

$$\bigl\{j : \bigl(\text{VIF}(\boldsymbol{\chi}_j) > 10\bigr) \text{ and } \bigl(\exists k \in \{1, \dots, r\} : \text{VIF}(\boldsymbol{\nu}_k) > 10 \ \&\ \text{corr}(\boldsymbol{\chi}_j, \boldsymbol{\nu}_k) \neq 0\bigr)\bigr\}\,.$$

4 Metrics

To evaluate the selected subset $\mathcal{A}$ we introduce criteria that estimate the quality of feature selection procedure. Variance Inflation Factor is the measure of multicollinearity in the matrix. We take the maximum VIF across all matrix columns:

$$\text{VIF} = \max_{j \in \mathcal{A}} \text{VIF}(\boldsymbol{\chi}_j).$$

The model stability is given by the logarithm of the ratio between minimal $\lambda_{\min}$ and maximum $\lambda_{\max}$ eigenvalue of the matrix $\mathbf{X}^T\mathbf{X}$:

$$R = \ln \frac{\lambda_{\min}}{\lambda_{\max}}.$$

The Root Mean Squared Error (RMSE) shows the quality of the model prediction. We estimate RMSE at the train and test data.

$$\text{RMSE}(\mathbf{Y}, \widehat{\mathbf{Y}}_{\mathcal{A}}) = \sqrt{\text{MSE}(\mathbf{Y}, \widehat{\mathbf{Y}}_{\mathcal{A}})} = \|\mathbf{Y} - \widehat{\mathbf{Y}}_{\mathcal{A}}\|_2, \quad \text{where} \quad \widehat{\mathbf{Y}}_{\mathcal{A}} = \mathbf{X}_{\mathcal{A}} \boldsymbol{\Theta}_{\mathcal{A}}^T.$$

Akaike Information Criteria (AIC) is a trade-off between prediction quality and the size of selected subset $\mathcal{A}$:

$$\text{AIC} = m \ln \left(\frac{\text{MSE}(\mathbf{Y}, \widehat{\mathbf{Y}}_{\mathcal{A}})}{m} \right) + 2|\mathcal{A}|.$$

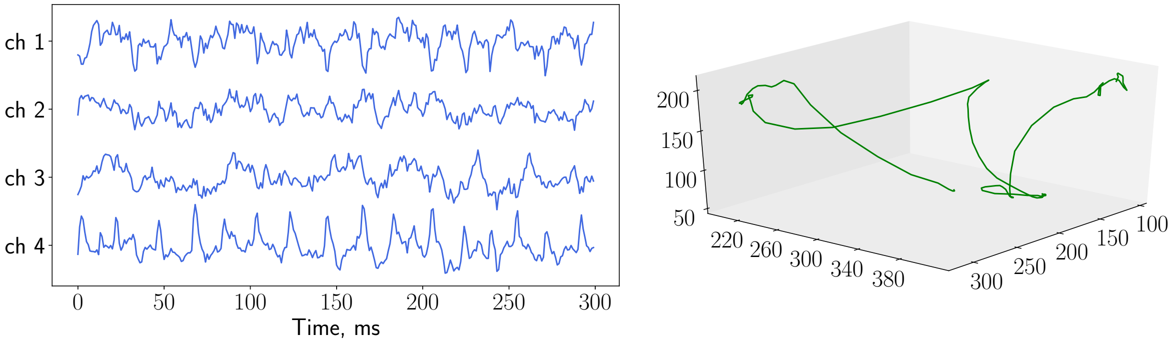


Figure 2: Brain signals and the corresponding hand position

5 Experiment

We carried out computational experiment with ECoG data from the NeuroTycho project. The input data consists of brain voltage signals recorded from 32 channels. The goal is to predict 3D hand position in the next moments given the input signal. The example of input signals and the 3D wrist coordinates are shown in Figure 2. The initial voltage signals are transformed to the spatial-temporal representation using wavelet transformation. The procedure of extracting feature representation from the raw data are described in details in [links]. Feature description in each time moment has dimension equals to 32 (channels) $\times$ 27 (frequencies) $=$ 864. Each object is the representation of local history time segment with duration $\Delta t = 1s$. The time step between objects is $\delta t = 0.05s$. The final matrices are $\mathbf{X} \in \mathbb{R}^{18900 \times 864}$ and $\mathbf{Y} \in \mathbb{R}^{18900 \times 3k}$, where k is a number of moments that we predict. We split our data into train and test parts with the ratio 0.67.

Figures 3 and 4 show the result of the QPFS algorithm, where we use the Relevance Aggregation strategy and $k = 1$. QPFS scores $\mathbf{a}_x$ decrease sharply. Only about one hundred features have scores significantly greater that zero. The test error stops to decrease using this one hundred features. It confirms that the initial data representation is redundant.

Figure 5 shows the dependencies in the matrices $\mathbf{X}$ and $\mathbf{Y}$. Some frequencies in the matrix $\mathbf{X}$ are highly correlated. The correlations between axes are not significant in comparison with correlations between consequent moments.

We apply QPFS algorithm with Relevance Aggregation strategy for different values of α_3 coefficient according to formulas (9). The dependence between target scores $\mathbf{a}_y$ with respect to α_3 for different values of k are shown in Figure 6. If we predict wrist coordinates only for one moment $k = 1$ targets scores are almost the same. It tells about the independence between x, y, and z coordinates. For $k = 2$ and $k = 3$ the scores of some targets becomes zero when α_3 increase.

We compare the proposed strategies of multivariate QPFS for the ECoG dataset.

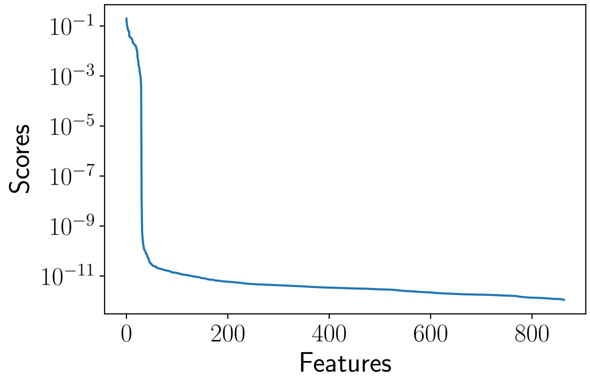


Figure 3: Sorted feature importances for the QPFS algorithm

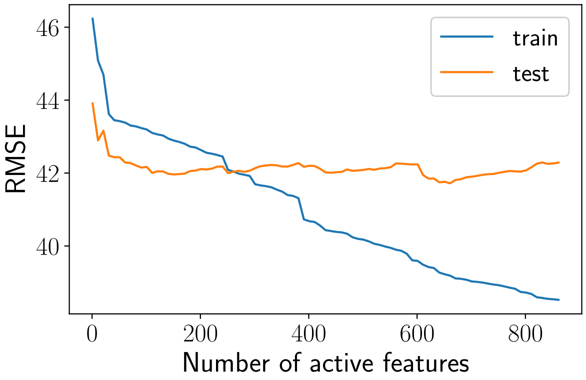


Figure 4: RMSE w.r.t. size of active set, features are ranked by QPFS algorithm

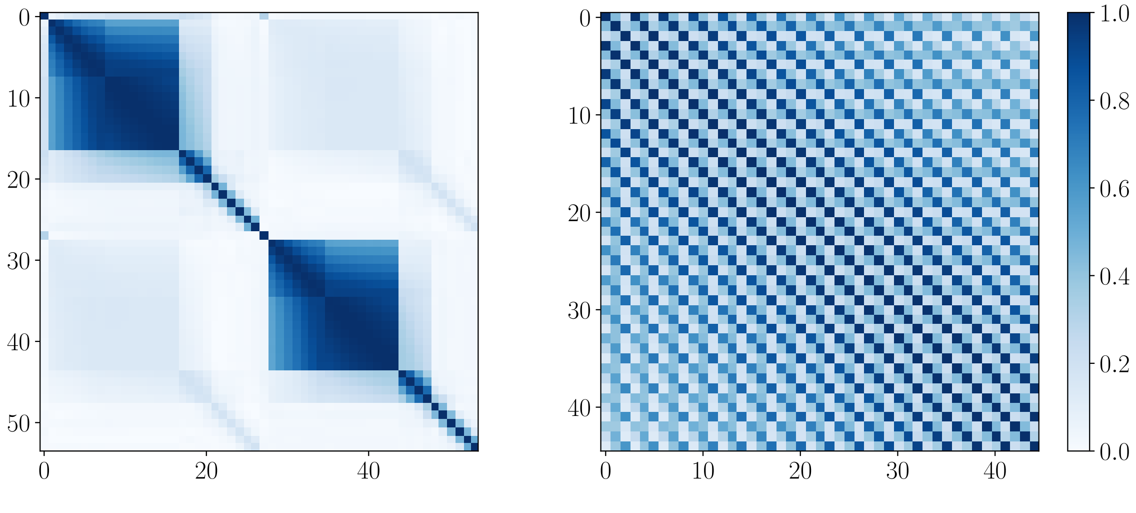
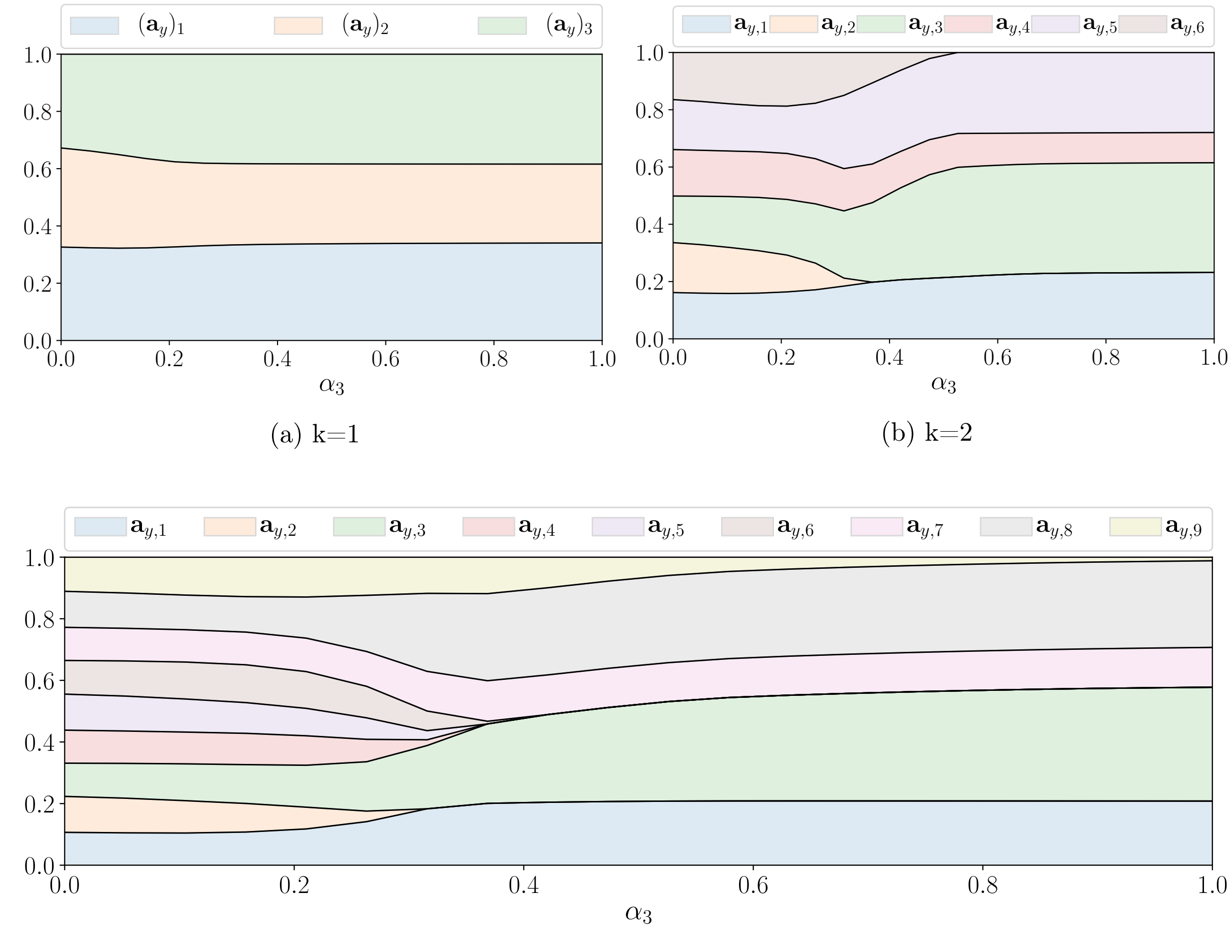


Figure 5: Correlation matrices for $\mathbf{X}$ and $\mathbf{Y}$



(a) k=1

(b) k=2

(c) k=3

Figure 6: Target importances $\mathbf{a}_y$ with respect to α_3 for QPFS with Relevance Aggregation

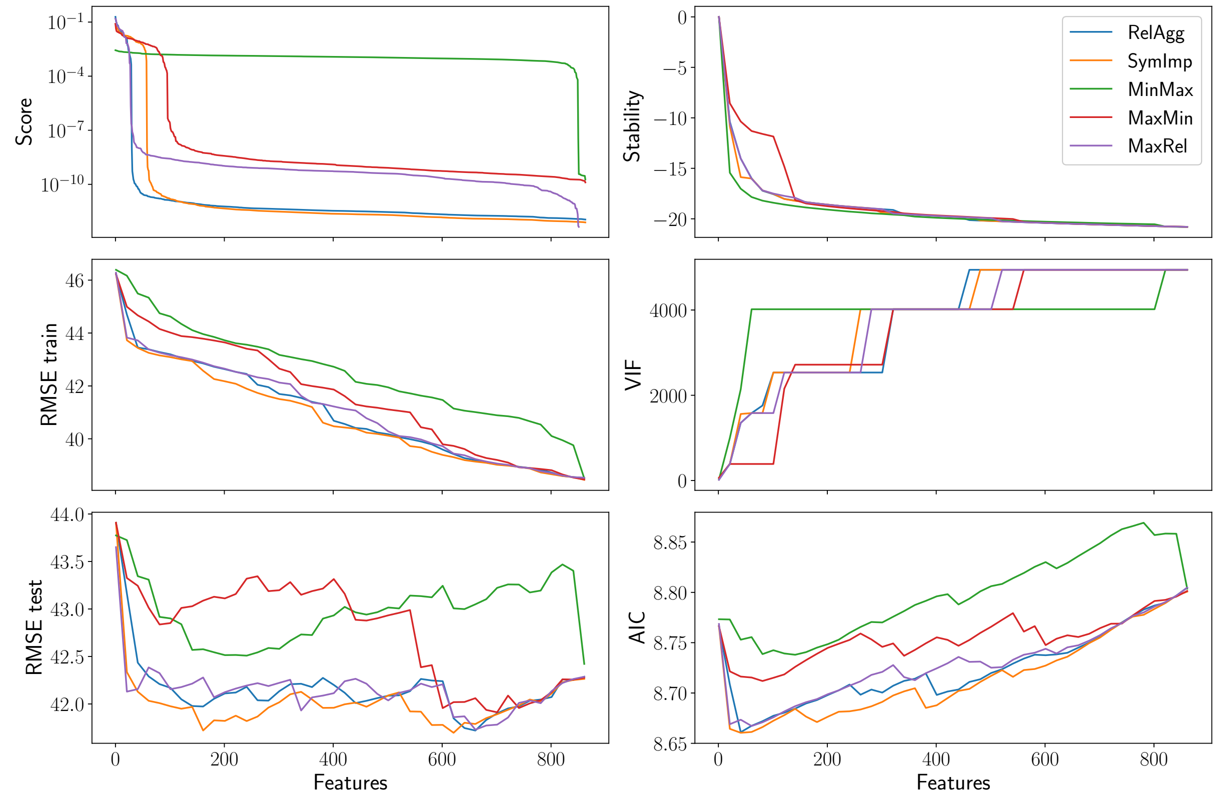


Figure 7: autoregression step = 1

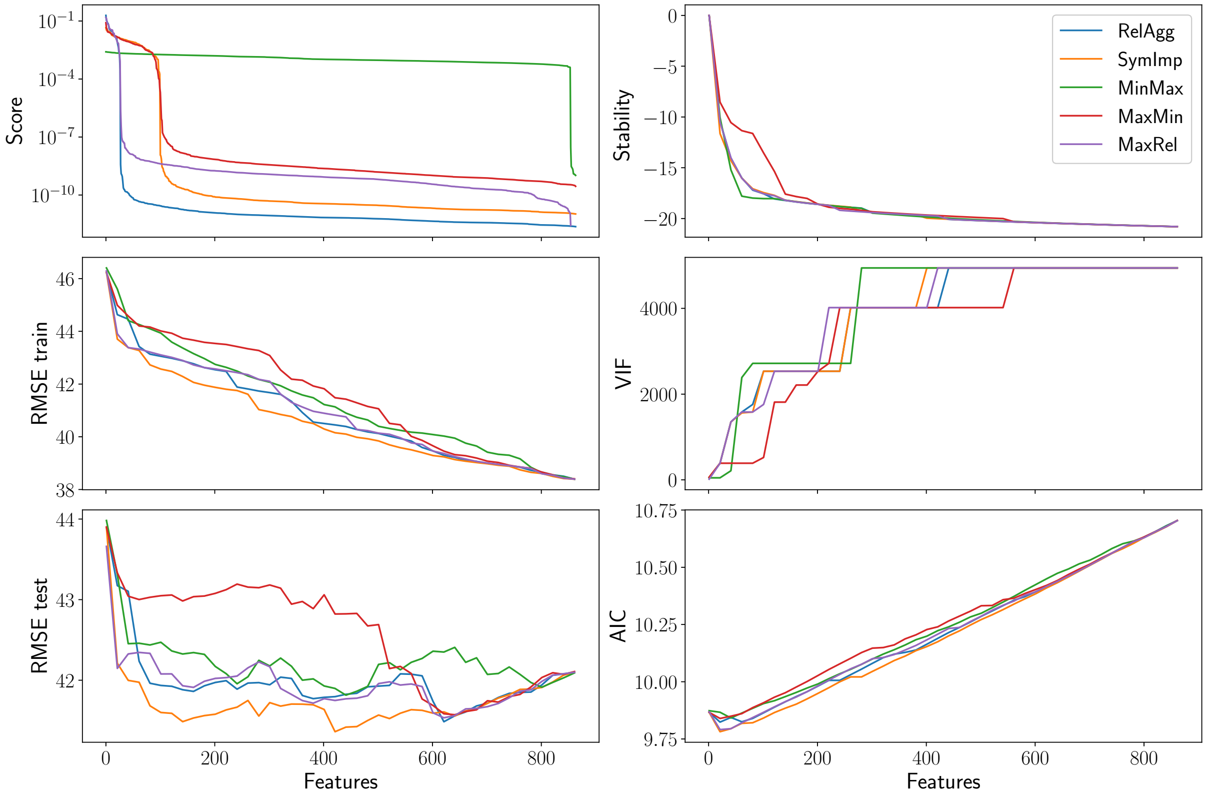


Figure 8: autoregression step = 3

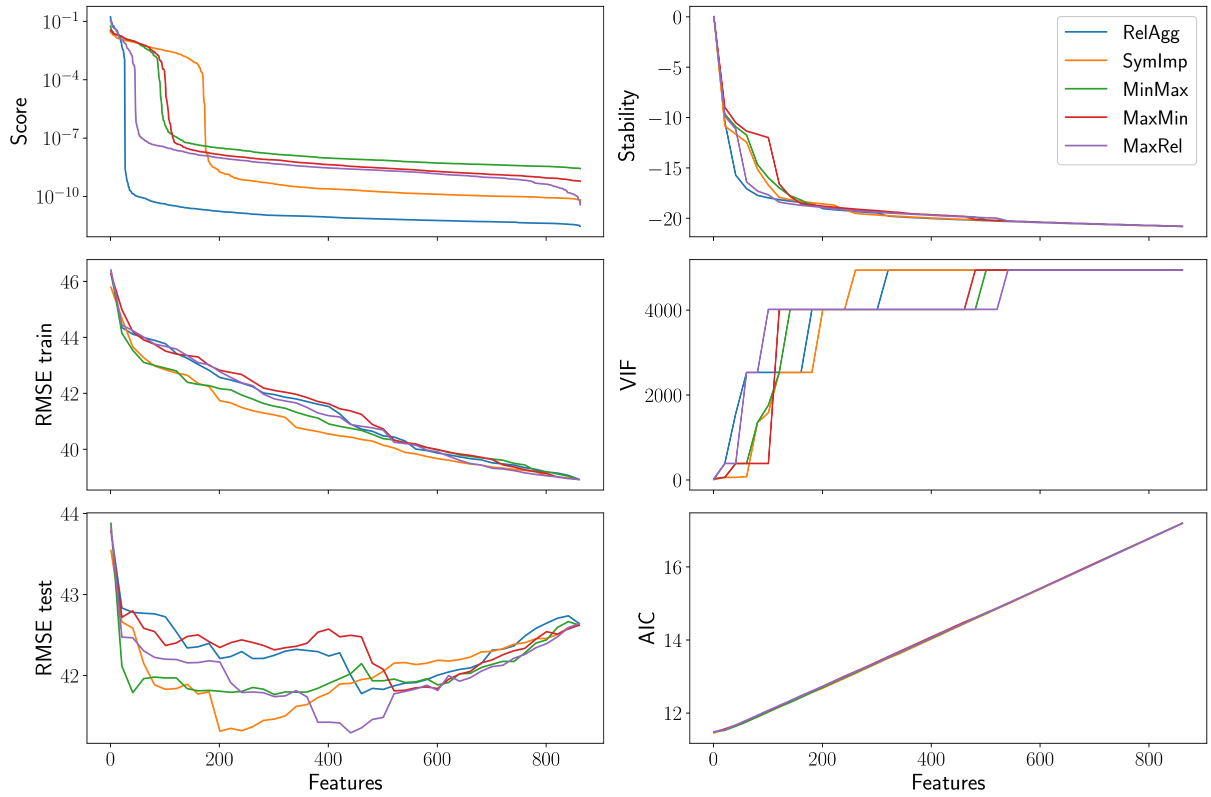


Figure 9: autoregression step = 15