Data-driven Gene Regulatory Network Inference based on Classification Algorithms

Sergio Peignier[a], Pauline Schmitt, Federica Calevro

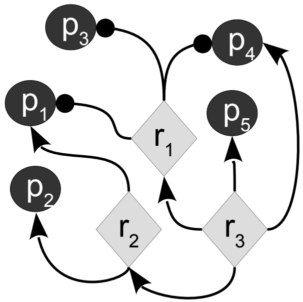
Université Lyon 1, INSA Lyon, INRA
BF2I, UMR0203, F-69621
Lyon, France

[a] sergio.peignier@insa-lyon.fr

Gene Regulatory Networks (GRNs)

Definition

Set of **interacting molecular regulators** (e.g. transcription factors) **controlling** the **creation** of **gene products** (e.g. mRNA, proteins).



- **Wide range** of **mechanisms**: (e.g., epigenetic, **transcriptional** ...)
- **Important biological role:**

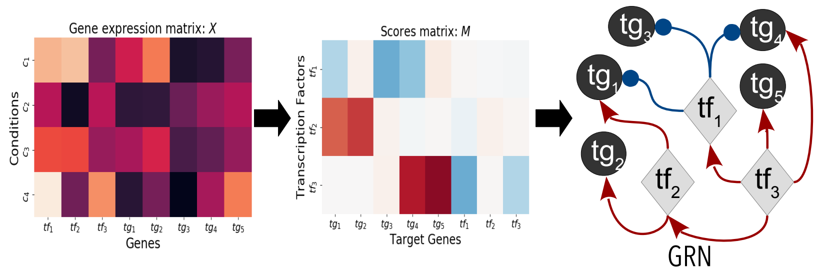
- Adaptation
- Versatility
- Differentiation
- Morphogenesis ...

Data-driven GRN Inference

General principle

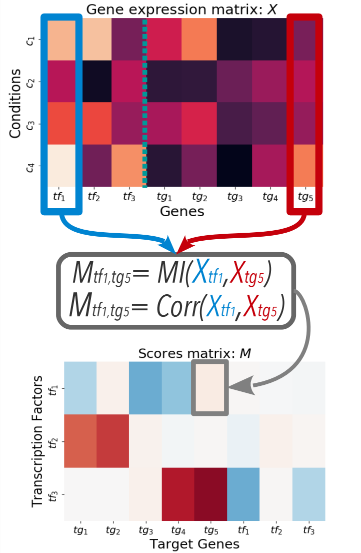
- **Based** on **high-throughput gene expression data**.
- **Score** possible **links** between:
  - **Regulators**, i.e. **Transcription Factors (TFs)**
  - **Target Genes (TGs)**
- **Select** most **promising links**.

Well-known paradigm: Simple, accurate, computationally efficient



Data-driven Inference Families

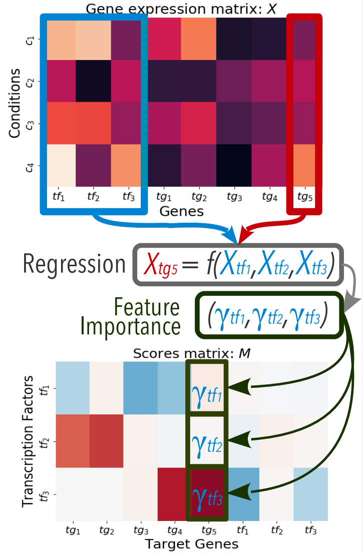
Correlation | Mutual Information



$$M_{tf_1,tg_5} = MI(X_{tf_1}, X_{tg_5})$$

$$M_{tf_1,tg_5} = Corr(X_{tf_1}, X_{tg_5})$$

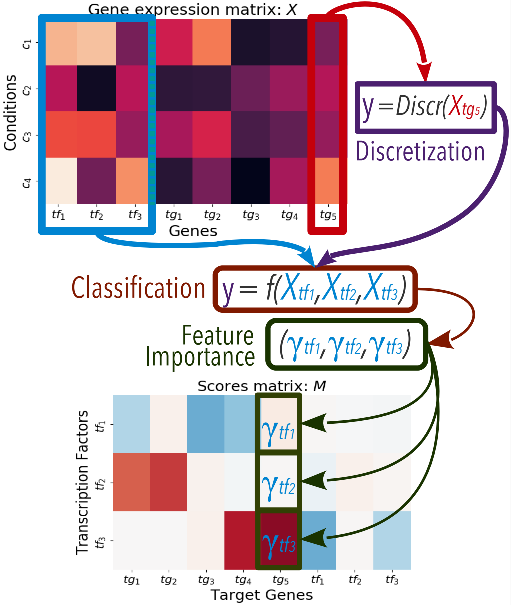
Regression Methods



Regression $X_{tg_5} = f(X_{tf_1}, X_{tf_2}, X_{tf_3})$

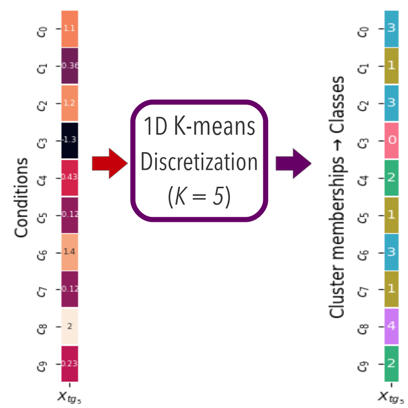
Feature Importance $(\gamma_{tf_1}, \gamma_{tf_2}, \gamma_{tf_3})$

Classification-based GRN Inference

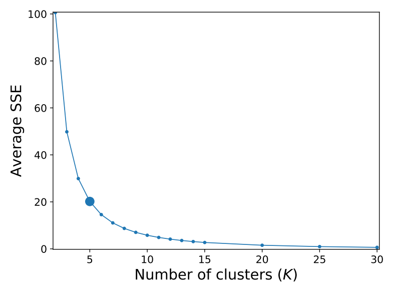


Target gene expression discretization

- K-means → **Discretize TG exp.**
- **Cluster membership** → **Class**

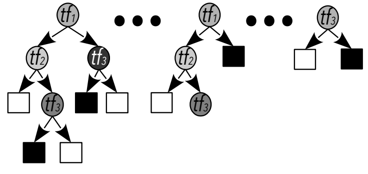


- **Avg. SSE** between **gene exp.** and **cluster centers** for different values of K.
- **Elbow** for $k = 5$ **clusters**.



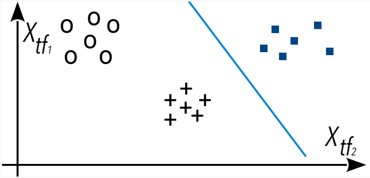
Classification Algorithms

Ensemble of decision trees[1]



- **R**andom **F**orest (**RF**)
- E**x**tremely **R**andomized **T**rees (**XRT**)
- **A**da**B**oost (**AB**)
- **G**radient **B**oosting (**GB**)

Support Vector Machine (SVM)[1]



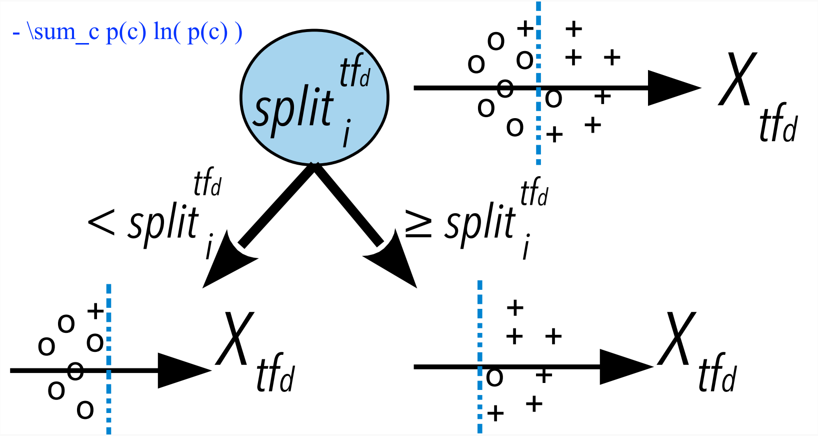
- One-vs-All linear multi-class **SVM**

[1] Implementations from **scikit-learn** (Python 3.7)

Feature Importance | Decision Tree based

Impurity gain (e.g., GINI, Entropy) for the i-th **split** along **feature** X_{tf_d}

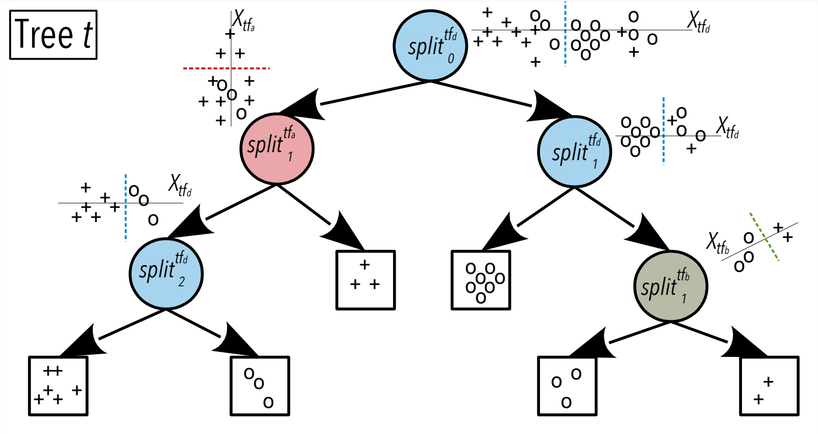
$$\boxed{\delta(split_i^{tf_d})}$$



Feature Importance | Decision Tree based

Importance of **feature** X_{tf_d} for tree t

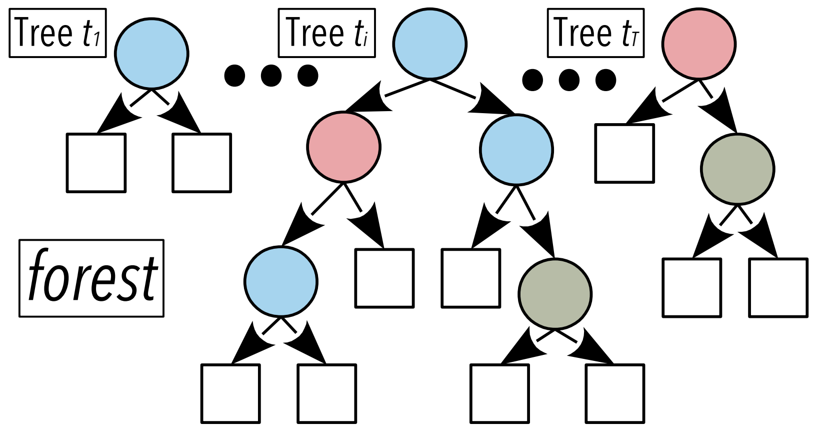
$$\boxed{\gamma_t^{tf_d} = \sum_i \delta(split_i^{tf_d})}$$



Feature Importance | Decision Tree based

Importance of **feature** X_{tf_d} for a *forest* , i.e., **set of trees**

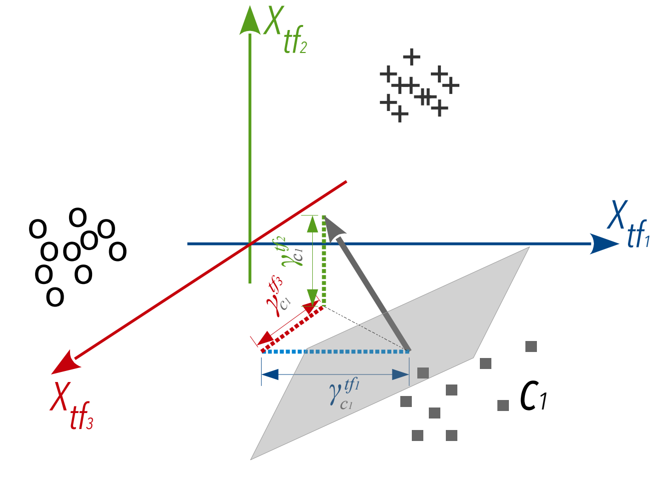
$$\boxed{\gamma^{tf_d} = \frac{\sum_{t \in forest} \gamma_t^{tf_d}}{|forest|}}$$



Feature Importance | One-vs-all linear SVM

$\gamma_c^{tf_d}$: **Importance** of **feature** X_{tf_d} for **class** c **linear SVM** →

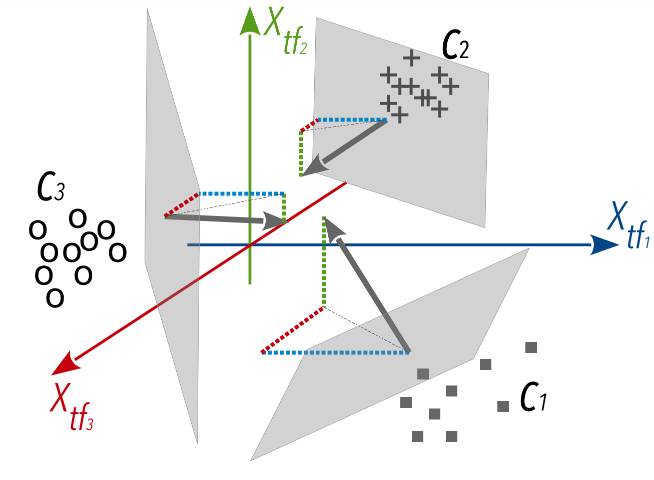
Norm of the **separating hyperplane orthogonal vector**



Feature Importance | One-vs-all linear SVM

Importance of **feature** X_{tf_d} for a **set** C of **one-vs-all linear SVM**

$$\boxed{\gamma^{tf_d} = \frac{\sum_{c \in C} \gamma_c^{tf_d}}{|C|}}$$



Evaluation Setup

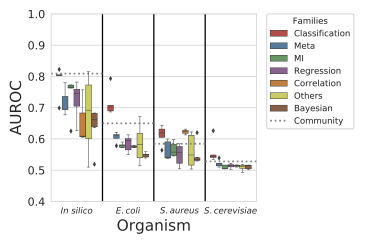
- **Evaluation protocol** designed by **[Marbach et al. 2012]**
- **Standard evaluation measures**: **AUROC** and **AUPR**
- **Impact** of **7 pre-processing techniques**.
- DREAM5 **Benchmark datasets**

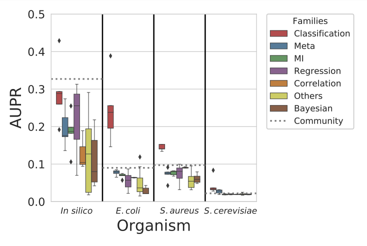
Dataset	# condit.	$|TGs|$	$|TFs|$
In Silico	805	1,643	195
S. aureus	160	2,810	99
E. coli	805	4,511	334
S. cerevisiae	536	5,950	333

- Comparison w.r.t. **36 methods**

Paradigm	# Methods
Community	1
MI	5
Meta	5
Regression	8
Correlation	3
Bayesian	6
Others	8

Results: Comparison with other Paradigms

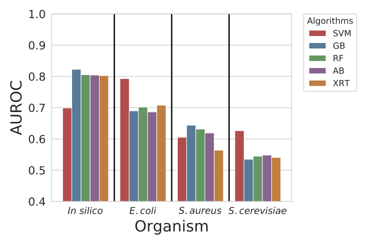


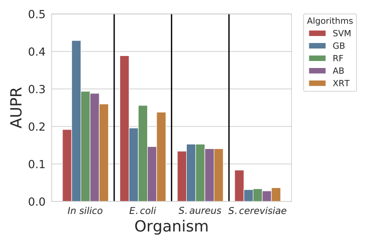


Paradigm	Avg. AUROC	Avg. AUPR
Classification	**0.67**	**0.18**
Community	0.64	0.13
Others	0.58	0.06
MI	0.60	0.09
Meta	0.60	0.09
Regression	0.59	0.09
Correlation	0.59	0.08
Bayesian	0.56	0.05

- **Best AUROC** and **AUPR** on avg.
- **Surpass community** results
- **Good results** for **all datasets**

Results: Classification Methods Comparison





- No ever-winning method.
- Analogous phenomenon reported in [Marbach et al. 2012]

Conclusion

Summary

- **Classification** methods **outperform** other **families** on avg.
- Interesting **complementary tool** for the **community**

Implementation

- **GReNaDIne Python package**:

Gene Regulatory Network Data-driven Inference

- **GitLab repository**:

`gitlab.com/bf2i/grenadine`

- **Documentation**:

`grenadine.readthedocs.io`

GReNaDIne

`pip install GReNaDIne`