Adaptive Feature Selection For Sparse Linear Bandits

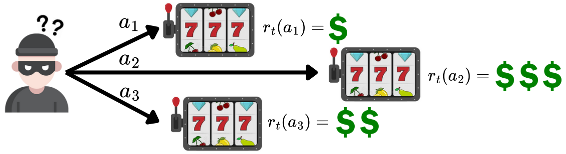
How can we improve algorithm performance by selecting a dynamic subset of features at every round?

Author: Martin Damyanov (mdamyanov@tudelft.nl) Supervisor: Julia Olkhovskaya

1. Background

Multi-armed bandits are a classic reinforcement learning problem where the ***exploration-exploitation dilemma*** is exemplified.

- Game that spans T rounds $t = 1 \dots T$
- At round t pick an action a_i^t
- Receive reward $r_t(a_i^t)$. Try to maximize total reward.



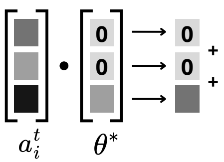
How is the reward generated?

- Environment picks an unknown parameter θ^*
- Rewards are dot product of action vector and θ^*

How do we measure performance?

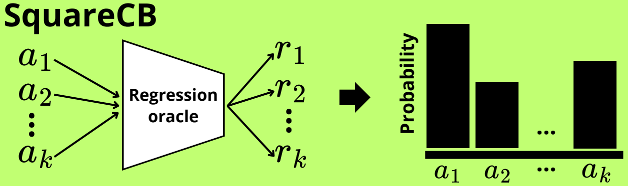
- Regret is the difference between our total reward and the best we could have done

When θ^* is sparse only a *small subset of the features will have an impact on the reward*. If we learn those features we can improve the regret bounds.



2. SquareCB

SquareCB

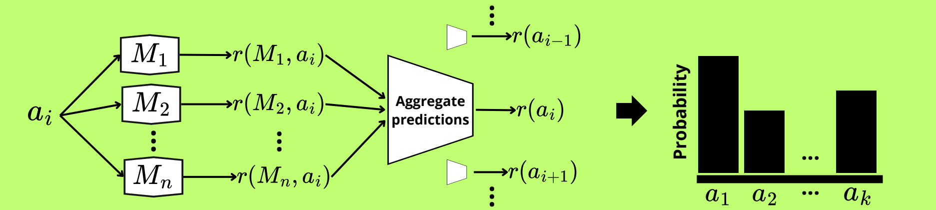


For every action predict the reward using linear regression

Create a distribution over the actions based on estimated reward and draw from it

- Use **different regression oracles** to test different feature selection strategies
- *Bayesian Regression with Feature Selection*
- *Feature Selection Sparse Linear Regression (FSSLR)*

4. FS-SCB - Expert advice for model aggregation



Define models that use different subsets of features

Predict reward for each model

Aggregate rewards. **Repeat for all actions.**

Define a distribution over the actions and draw from it

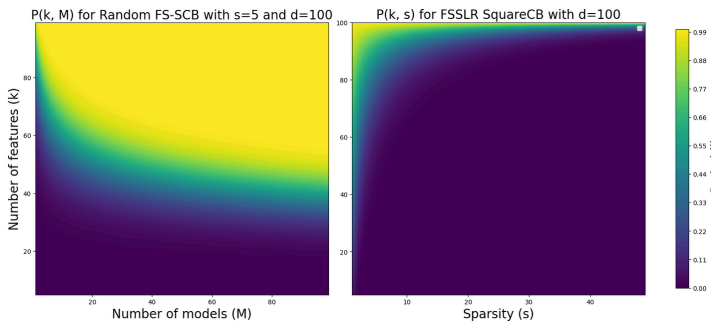
It is important how we create the models. We consider two strategies.

- **Random Feature Selection**
- **Bayesian Feature Selection**

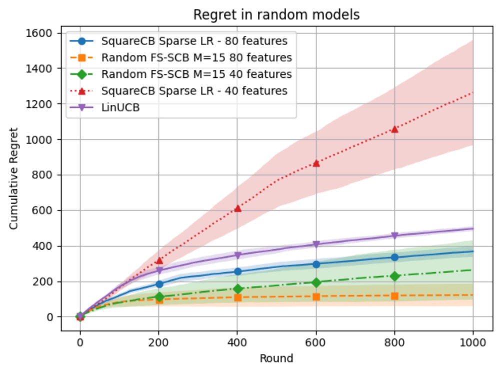
5. Experiments & Results

Relative comparison of all algorithm's performance

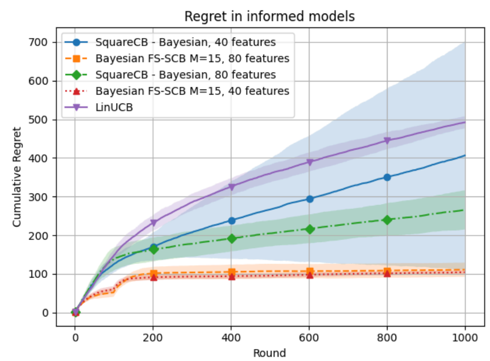
Algorithm	Performance on large feature subsets	Performance on small feature subsets	Resource consumption
SquareCB + FSSLR	Good	Poor	Low
SquareCB + Bayesian Feature Selection	Good	Medium	Medium
FS-SCB + Random Model Selection	Good	Medium	High
FS-SCB + Bayesian Model Selection	Very Good	**Excellent**	Very high



Regions in the hyperparameter space where the algorithms have a high probability of capturing the correct subset of features.



Comparison of algorithms with random selection strategies. Reducing the number of features drastically reduces the performance



Comparison of algorithms with Bayesian Feature Selection. Smaller feature subsets do not have a big impact on the performance.

6. Conclusions

- I implemented two bandit algorithms with two feature selection strategies
- In both cases, **random** strategies were clearly outperformed by more **sophisticated** alternatives
- Treating models as expert advice allows for **imperfect selection strategies** to have much **better performance**

References for images

Slot machines: https://www.dreamstime.com/slot-machine-simple-design-isolated-white-background-flat-image181537574
Dollar Sign: http://www.rawshorts.com
Bandit: https://www.flaticon.com/free-icon/bandit_4751274