Improving Individual Learning through Performance Tracking

Mihaela van der Schaar
University of Oxford
Alan Turing Institute

Joint work with **Jie Xu (U Miami)**

Personalized Education

- Trend in education: larger and larger classes
  - physical classrooms
  - MOOCs
- Unsatisfactory because students are heterogeneous
  - heterogeneous backgrounds & abilities
  - heterogeneous styles of learning
  - heterogeneous goals

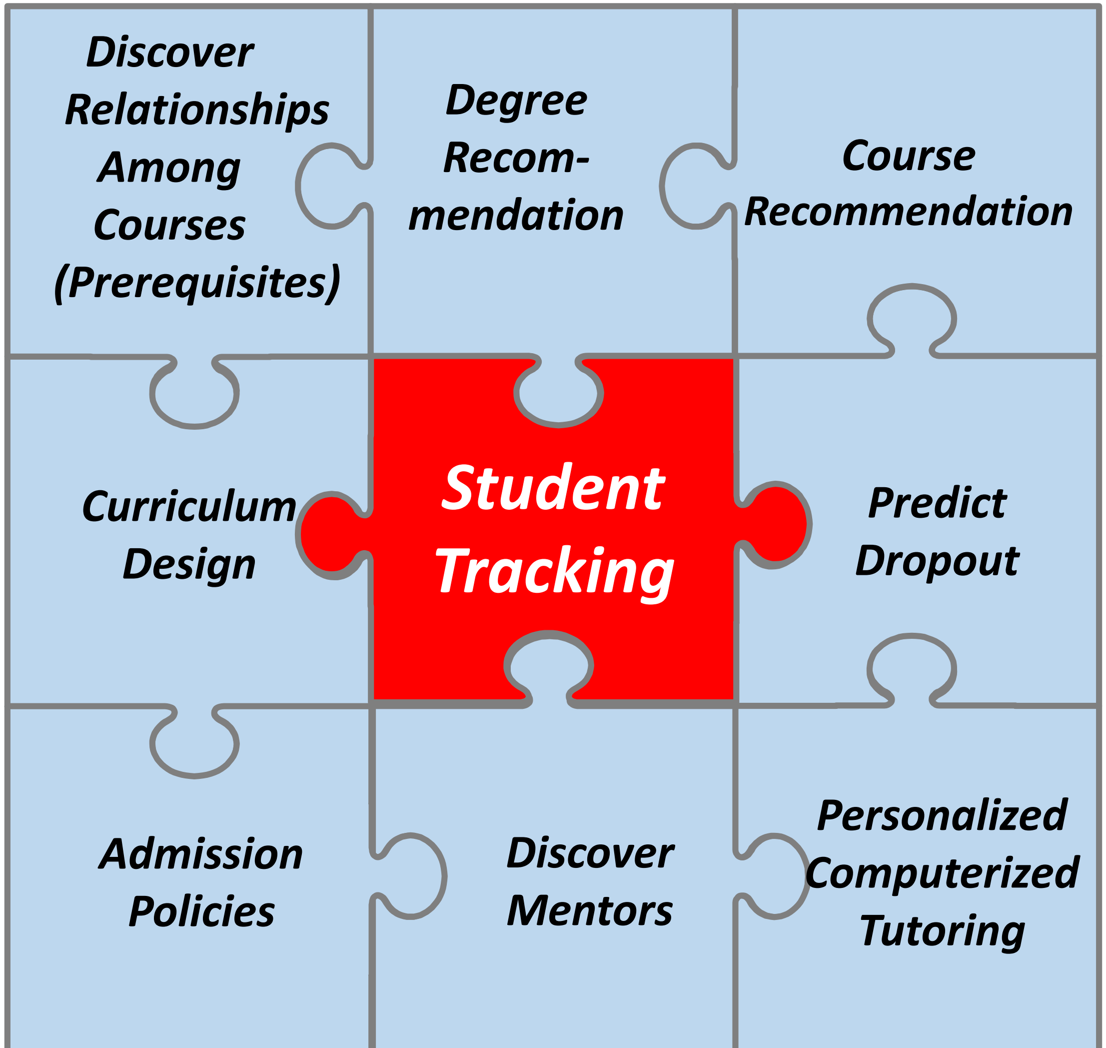
=> **Personalization**

- maintain engagement
- improve learning

- **Our approach: electronically personalized interactive environment (EPIE) for each student**

  => "as if" one mentor for every student

EPIE



http://medianetlab.ee.ucla.edu/EduAdvance

Some facts

- **Students do not graduate on time!**
  - Only 50 out of 580+ public 4-year institutions in the US have on-time graduate rates greater than 50%
- **Time is money**
  - 1 extra year of a public 4-year college = $22,826 in year 2014
- **Student loan debt > a trillion dollars**
  - More than USA's combined credit card and auto load debts!

- System that can *continuously* track students' performance and *accurately predict* their future performance
- *Timely* identification of students unlikely to graduate on time (and/or with a decent GPA)
- Enables timely interventions, course recommendations etc.

Challenges

- Students heterogeneity
  - In backgrounds, chosen areas (majors), selected courses and course sequences
  - How to handle heterogeneous student data?
- Not all courses are created "equal"
  - How to discover the underlying relationships existing among courses and use this for student tracking and course recommendations?
- Sequential prediction problem
  - Continuous tracking of student learning and student performance
  - How to incorporate the evolution of student progress into performance prediction?

Model

Student i

- *Static features*: background $\theta_i \in \Theta$
  - High school GPA, SAT scores etc.
- *Dynamic features*:
  - x_i^t - performance/grades at the end of term t
  - $x_i^1, x_i^2, \ldots, x_i^t$ quantifies the student's performance across time

Goal

- **Predict final cumulative GPA after each term t**

$$\widehat{GPA}_i^t = \frac{\sum_{j \in \bar{S}^t} c(j) x_i(j) + \sum_{j \in J \setminus \bar{S}^t} c(j) \hat{x}_i(j)}{\sum_{j \in J} c(j)}$$

  - J: set of all courses
  - $\bar{S}^t$: set of courses completed by term t
  - $c(j)$: course credit
  - $x_i(j)$: grade for completed courses
  - $\hat{x}_i(j)$: predicted grade for uncompleted courses

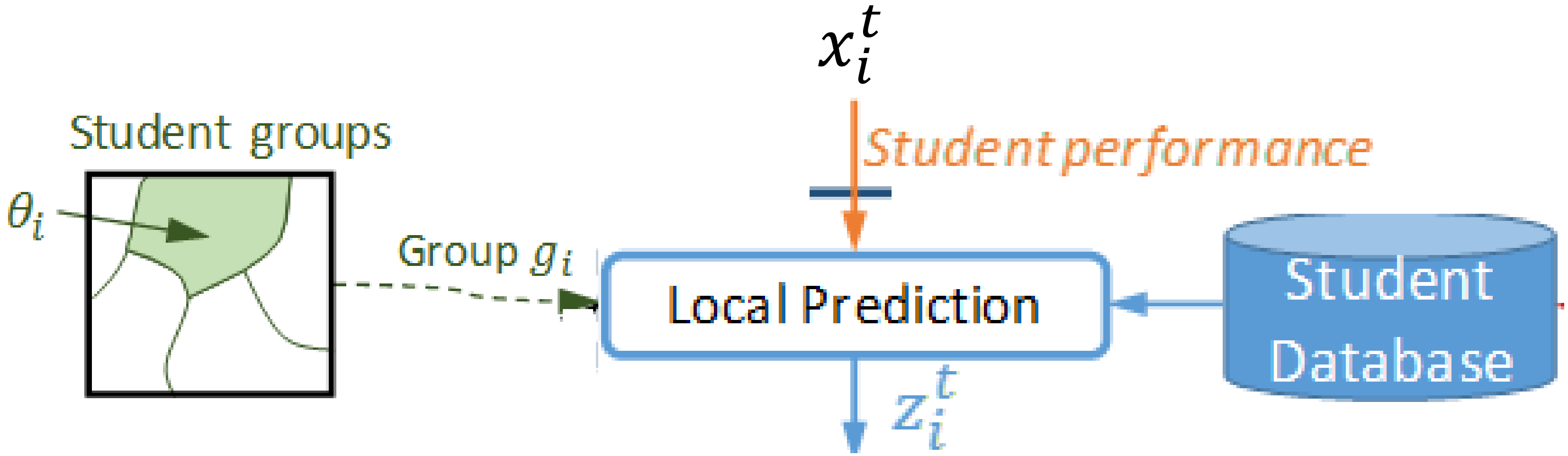
- Related objective: predict the grade for each uncompleted course

Proposed solution: hierarchical approach

Base layer

- A set of base (local) predictors H^t implemented using different prediction algorithms
- Each base (local) predictor $h \in H^t$ outputs

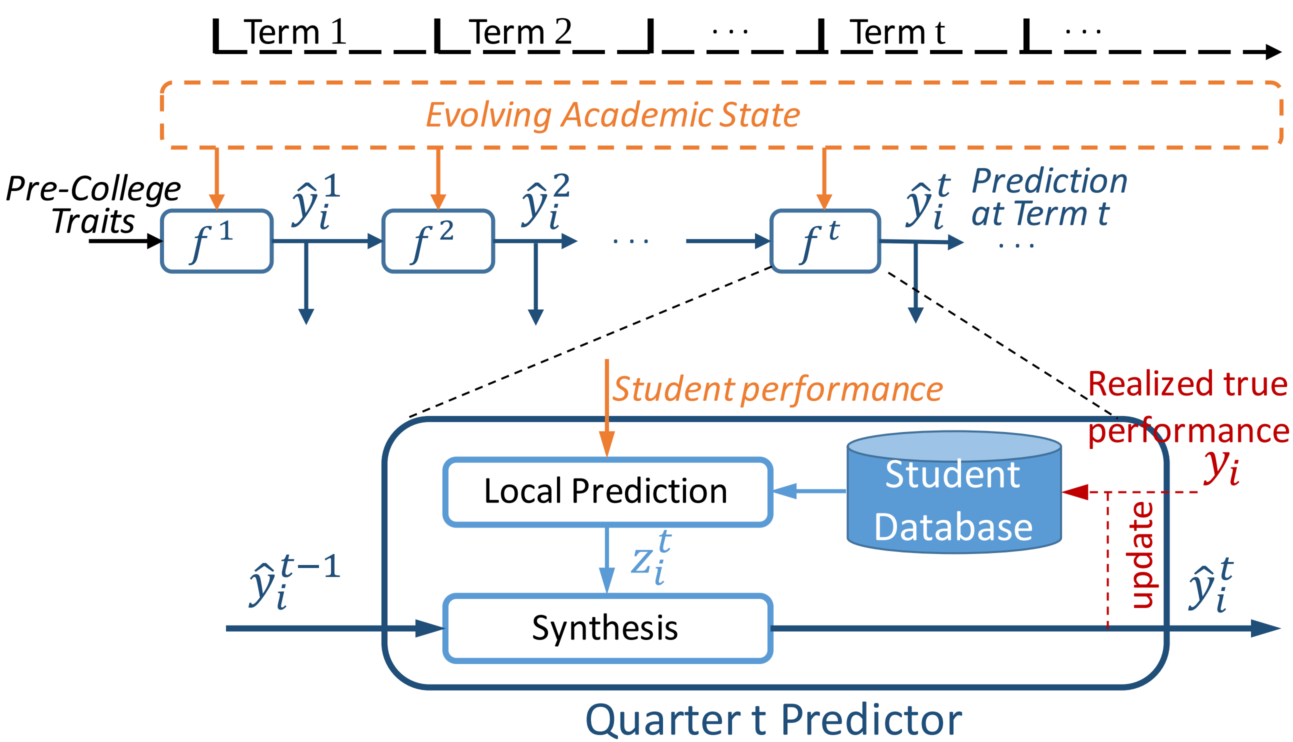
$$z_{h,i}^t = h\left(\theta_i, x_i^t\right)$$



Proposed solution: hierarchical approach

Ensemble layer

- One ensemble predictor f^t for each term t
- Each f^t *synthesizes* output $\hat{y}_i^{t-1}$ of previous ensemble predictors & base predictors $z_{h,i}^t$ and *outputs* $\hat{y}_i^t$



Design questions

- How to construct the base predictors?

- Customize to grade prediction

- How to construct the ensemble predictors?

- Consider temporal correlation

Learning Base Predictors

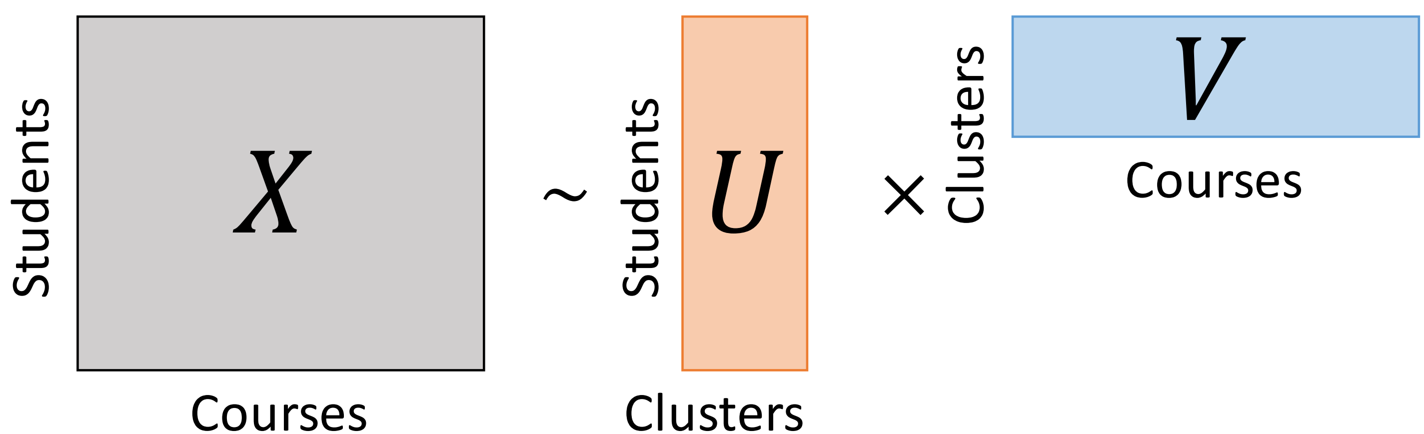
- An important question when training h^t: how to construct the input feature space
  - Using all courses increases complexity and adds noise

- Idea: learn the courses that are most relevant to the course for which we need to issue a prediction

Learning Relevant Courses

- A matrix X of size $I \times J$
  - Rows represent students
  - Columns represent courses

- We aim to find course clusters by factorizing $X = U^T V$
  - U is the compressed grade matrix of size $K \times I$
  - V is the course-cluster matrix of size $K \times J$
  - K is the number of course clusters that we try to find



Challenge

- Student grade matrix X can be sparse since it is constructed using data from multiple study areas and students only take a subset of courses

- Difficult non-convex optimization problem - cannot be solved using standard SVD implementations

- Use ***probabilistic matrix factorization*** method in [R. Salakhutdinov and A. Mnih, NIPS 2011]

Learning Relevant Courses

- Once U and V are found
  - Method 1: course j is assigned to a single cluster k with the highest value among all possible course clusters
$$k(j) = \arg\max_k V_{k,j}$$
  - Method 2: course j belongs to cluster k if $V_{k,j} > \bar{v}$, where $\bar{v}$ is a predefined threshold value.

- For term t base predictor h^t
  - only relevant courses that have been taken by term t are used for training h^t

Learning Ensemble Predictors

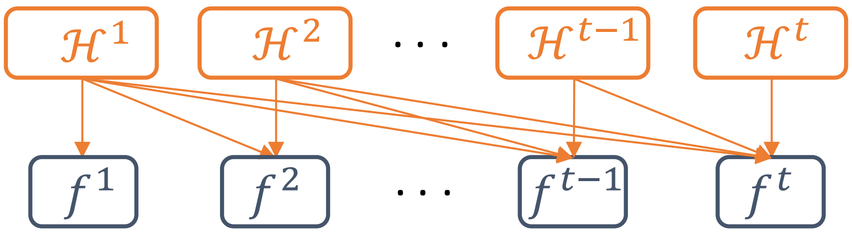
- A stochastic setting
  - Students arrive in sequence $i = 1,2, \ldots$
  - Suitable for both offline training and online updating
- Students are assigned to clusters based on static feature θ_i

- In each term t
  - Each base predictor $h^t \in H^t$ makes a prediction $z_{h,i}^t = h^t(\theta_i, \tilde{x}_i^t)$
    - $\tilde{x}_i^t$ is performance state restricted to the relevant courses
  - A total number of $t \times H$ prediction results by term t

- Goal: synthesize base predictions to output final prediction

Some Possible Synthesis Methods

- **Directly** utilizing all past information

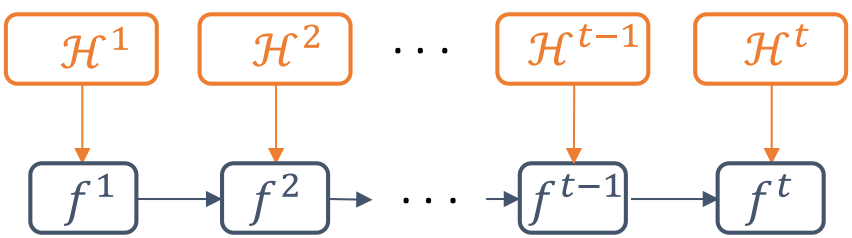


\# of inputs at term t

$t \times H$

Large when t is large
Treat info equally

- **Progressively** utilizing past information



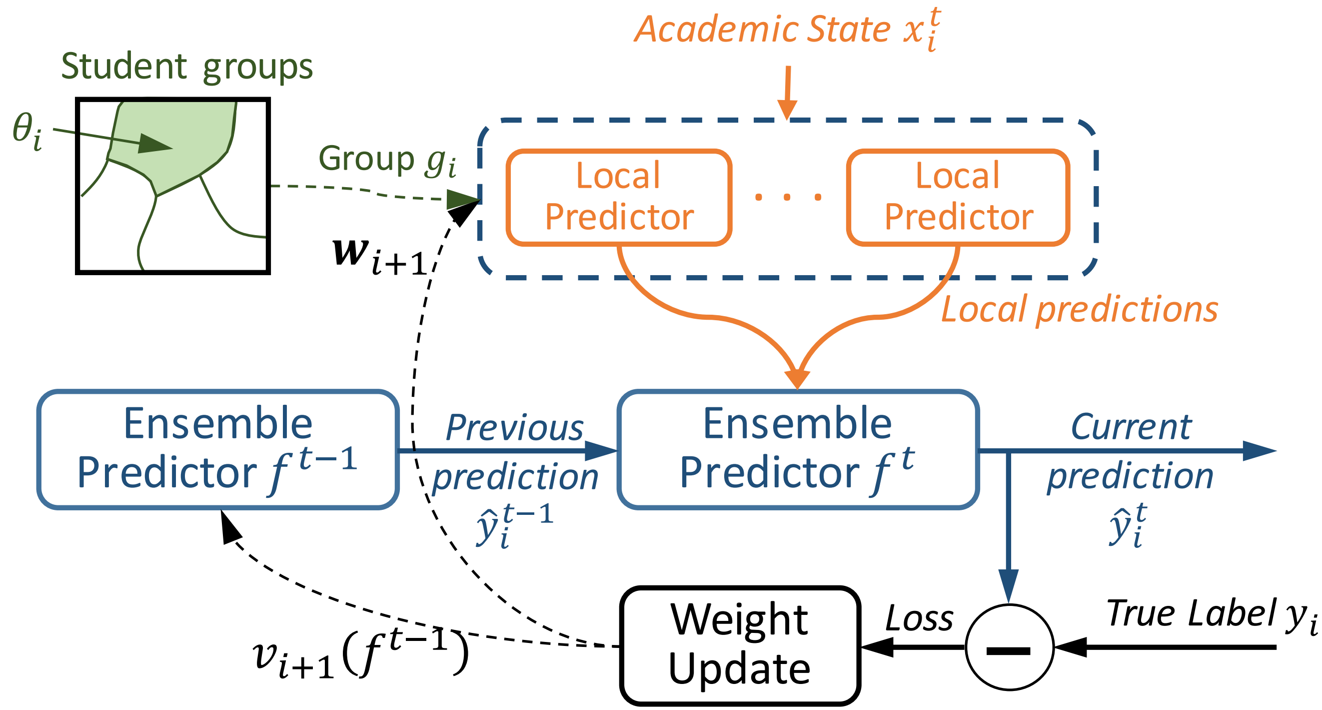
$H + 1$

Constant, independent of t
Automatically discounts old info

Progressive Prediction

Exponentially weighted average forecaster

- $w_i(h^t)$: weight for base predictor h^t
- $v_i(f^t)$: weight for ensemble predictor f^t
- Final prediction: $\hat{y}_i^t = \frac{\sum_{h \in H^t} w_i(h) z_{i,h}^t + v_i(f^{t-1}) \hat{y}_i^{t-1}}{\sum_{h \in H^t} w_i(h) + v_i(f^{t-1})}$



Progressive Prediction

Exponentially weighted average forecaster

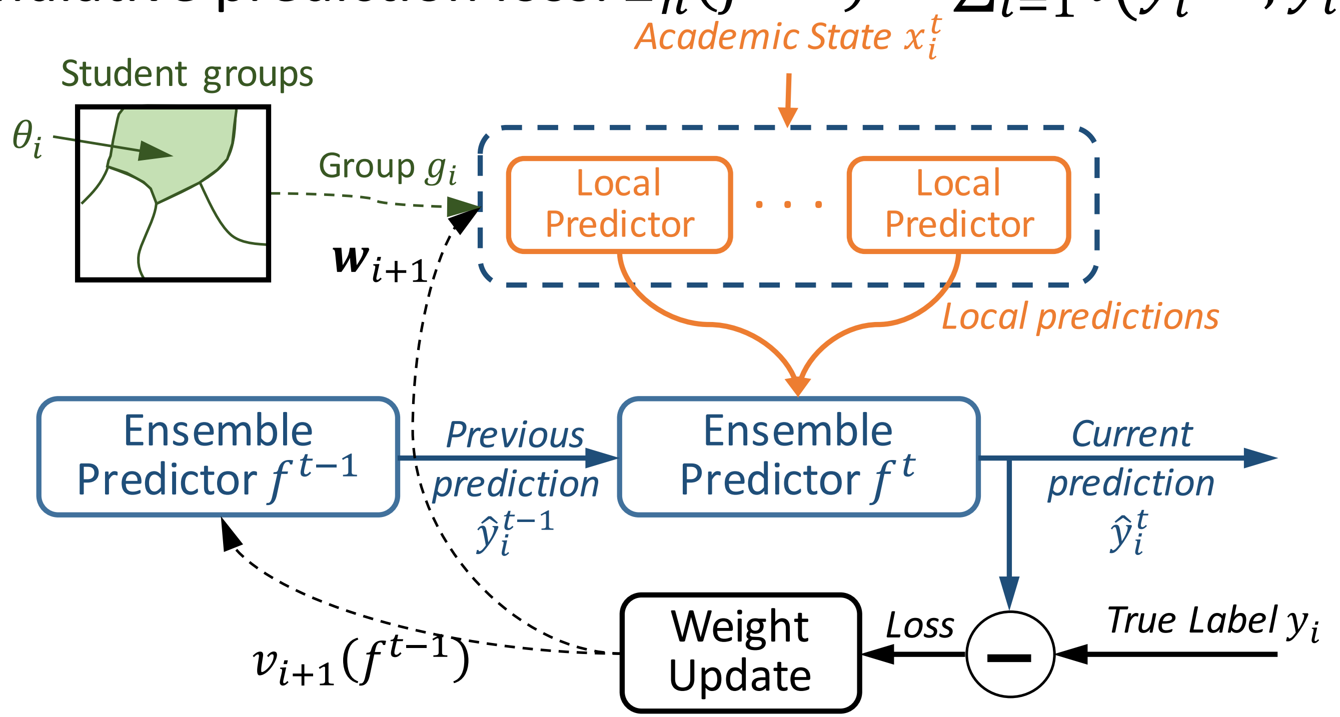
- Weights are updated according to their cumulative prediction loss

$$w_{i+1}^{t}(h^t) = \exp\left(-\eta_i L_i(h^t)\right)$$

  - Cumulative prediction loss: $L_n(h) = \sum_{i=1}^{n} l(z_{i,h}^{t}, y_i)$

$$v_{i+1}^{t-1}(f^{t-1}) = \exp\left(-\eta_i L_i(f^{t-1})\right)$$

  - Cumulative prediction loss: $L_n(f^{t-1}) = \sum_{i=1}^{n} l\left(\hat{y}_i^{t-1}, y_i\right)$



Performance

Learning regret up to student n

$$\mathrm{Reg}^t(n) = L_n(f^t) - L_n^{*,t}$$

$L_n^{*,t}$ is best local prediction performance in hindsight

Theorem:

Regret is sublinear in n

$\mathrm{Reg}^t(n) < O(\sqrt{n})$

Corollary:

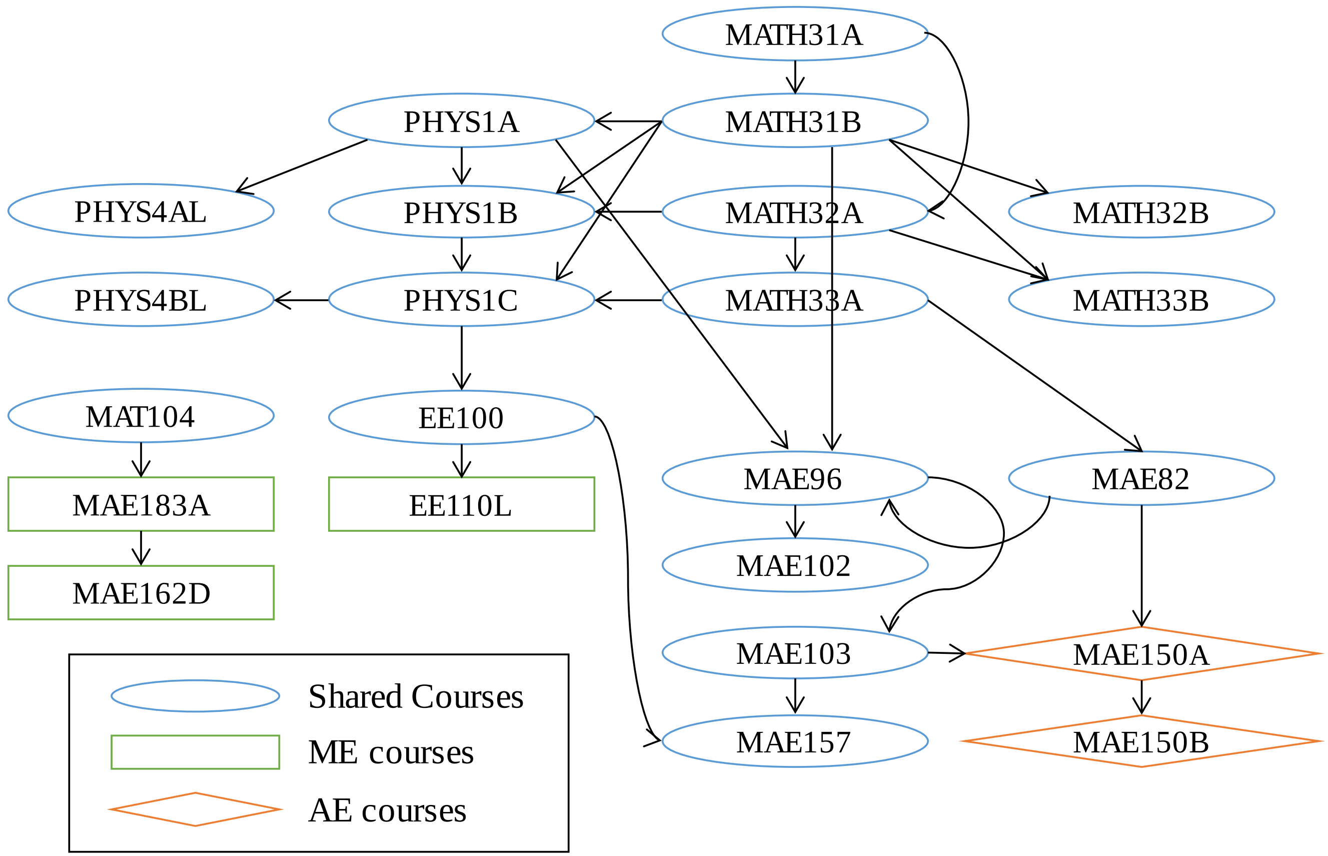
$\lim_{n\to\infty} \frac{1}{n} \mathrm{Reg}^t(n) \to 0$: asymptotically optimal

Performance

- The direct method has an expected regret bound
$$E[\mathrm{Reg}^t(n)] \leq O\left(\sqrt{n \ln(Ht)}\right)$$

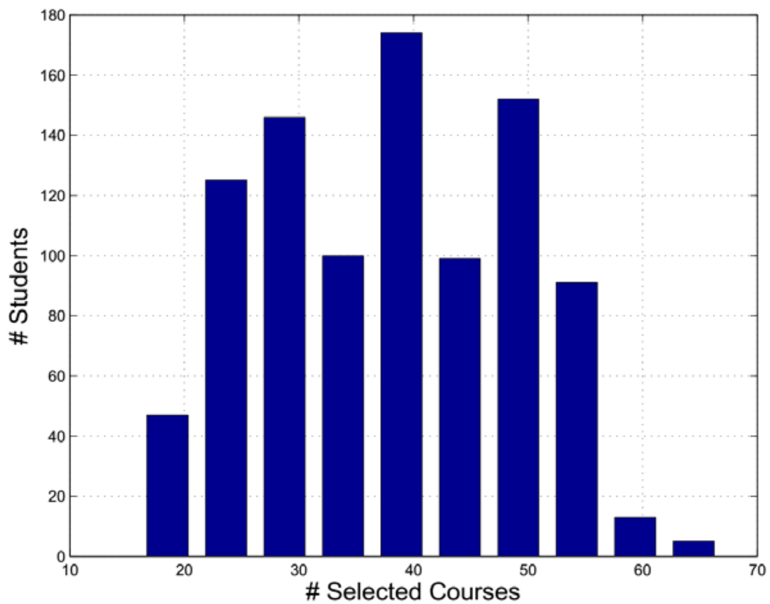
Dataset

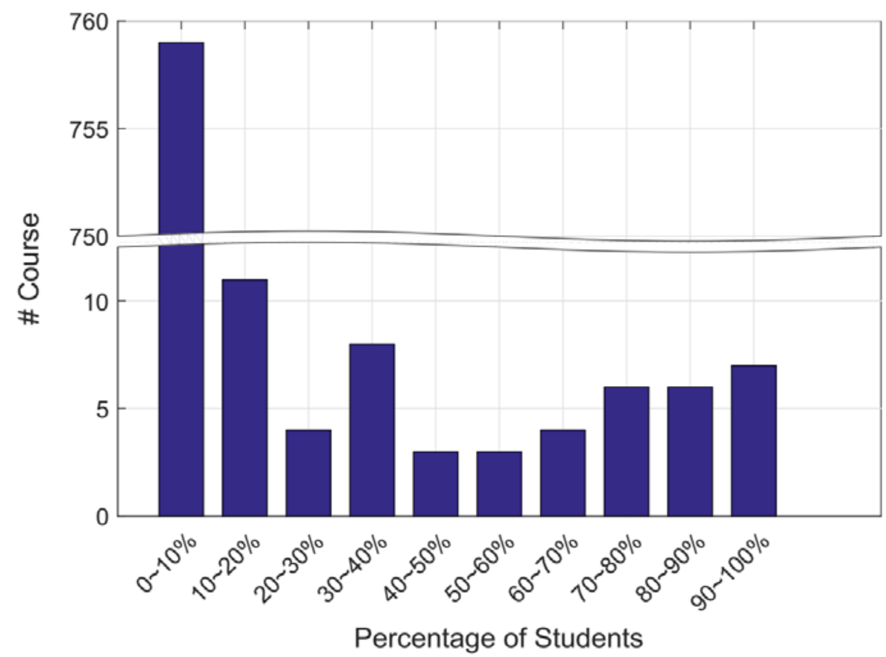
- 1169 anonymized undergraduate students in UCLA Mechanical and Aerospace Engineering department



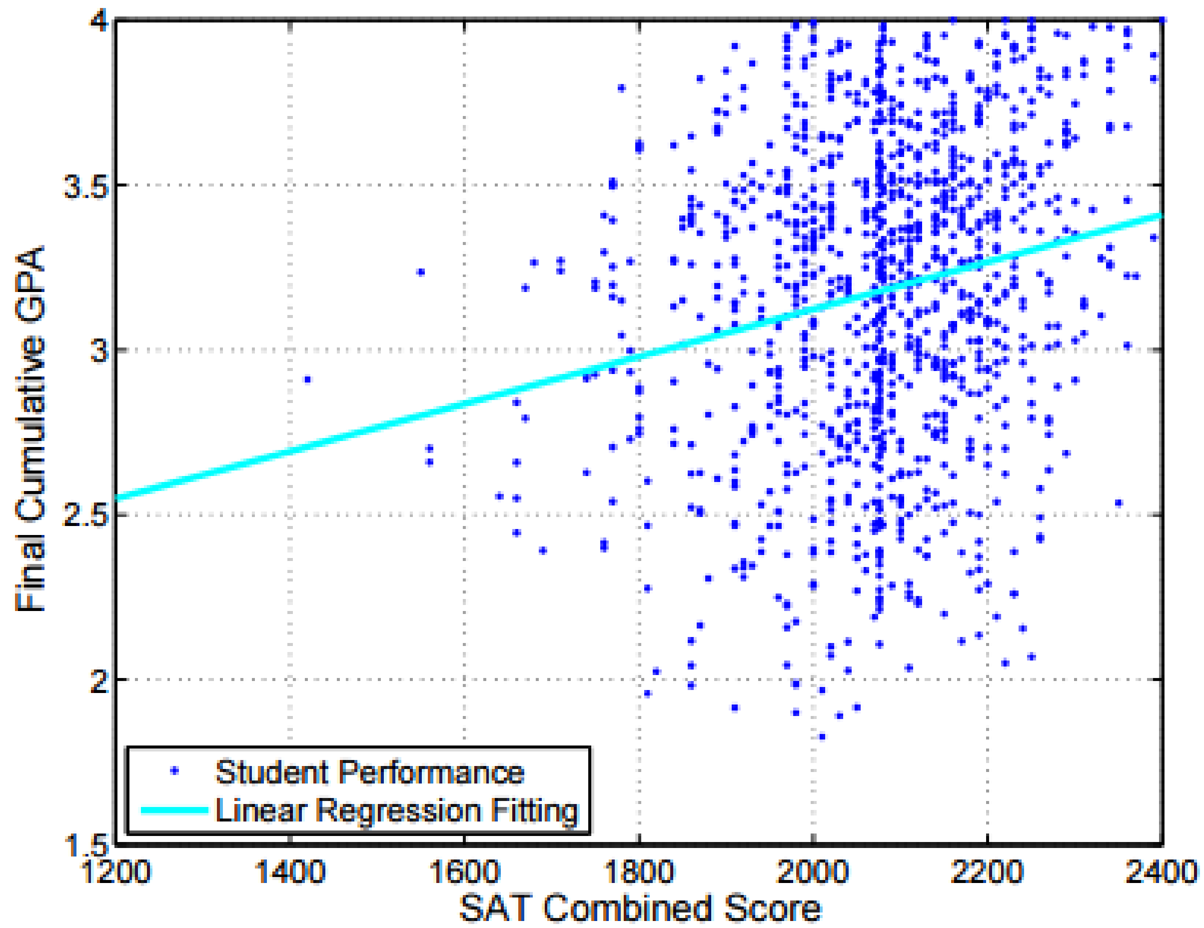
Dataset

- Selected Courses
  - Average number of courses is 38
  - Total number of distinct courses is 811.
  - 759 of them are taken by less than 10% of the students

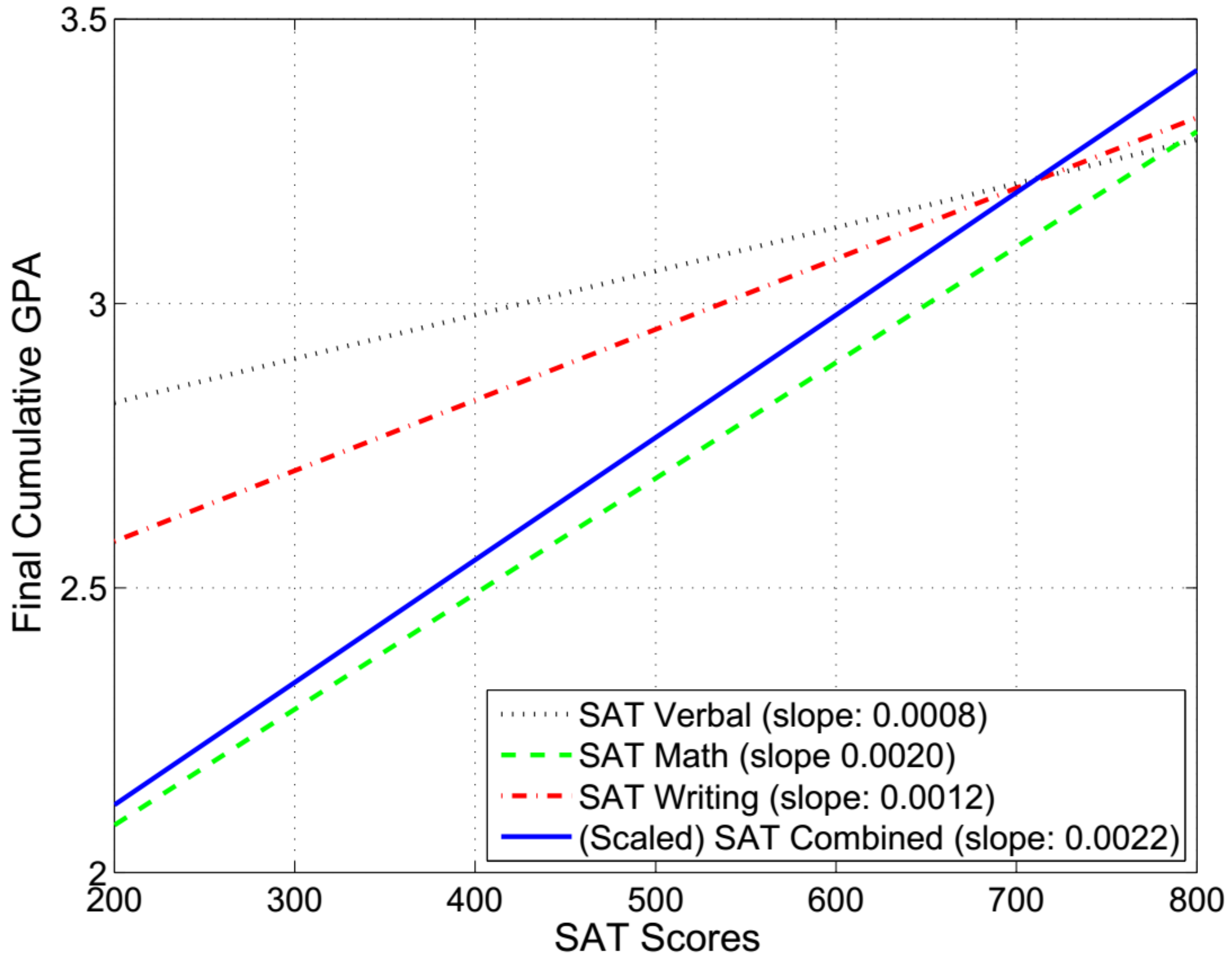




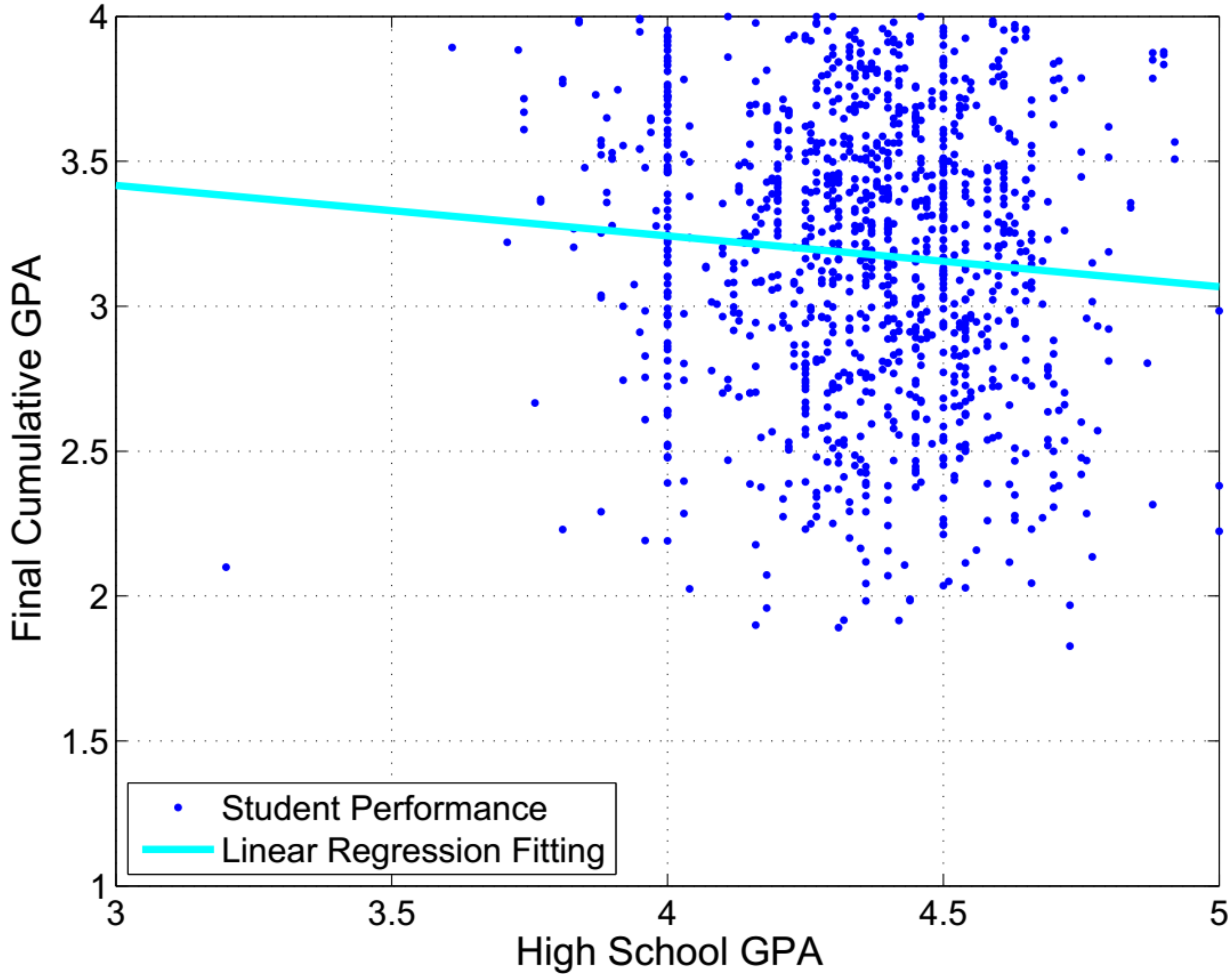
Finding 1: Students with higher SAT also obtain higher final GPA



Finding 2: SAT Math is better predictor, compared with Verbal and Writing

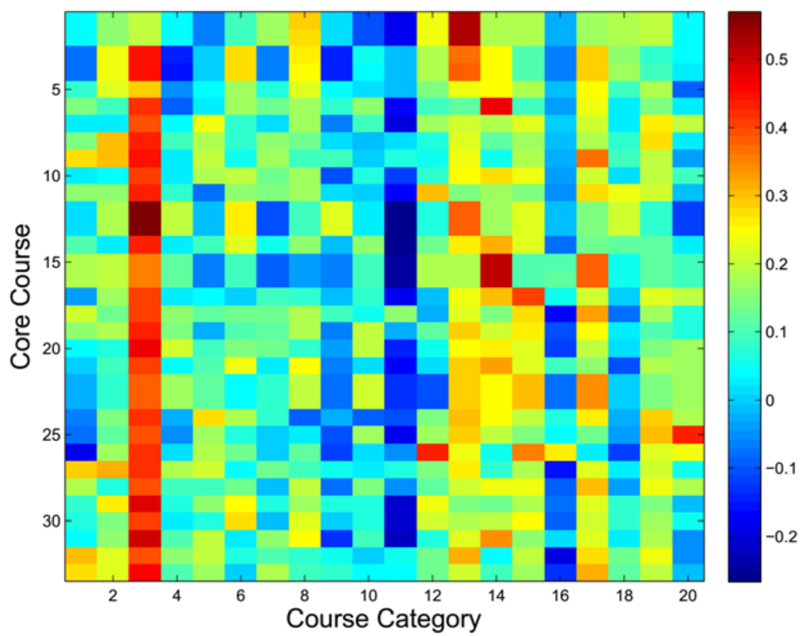


Finding 3: Students' high school GPA is almost *not correlated* with final GPA

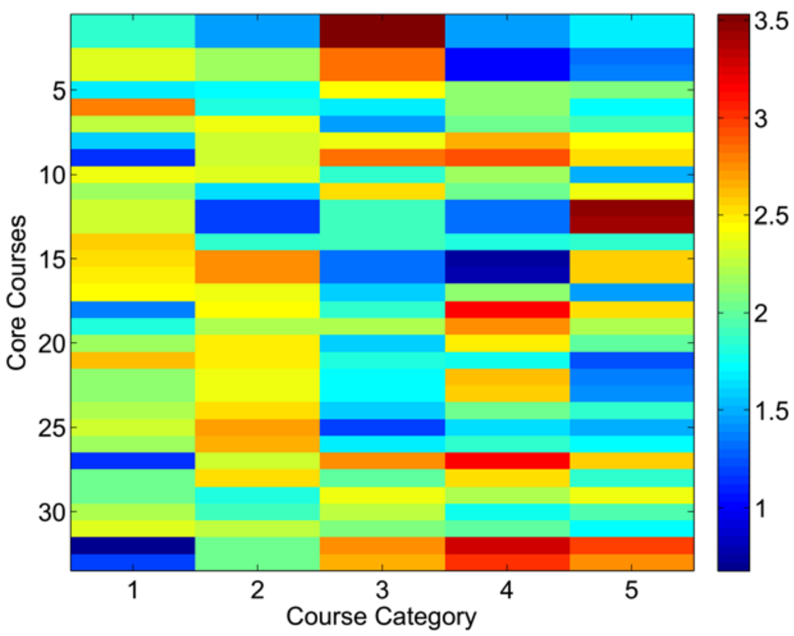


Correlated Courses

- Matrix factorization results (K = 20, K = 5)



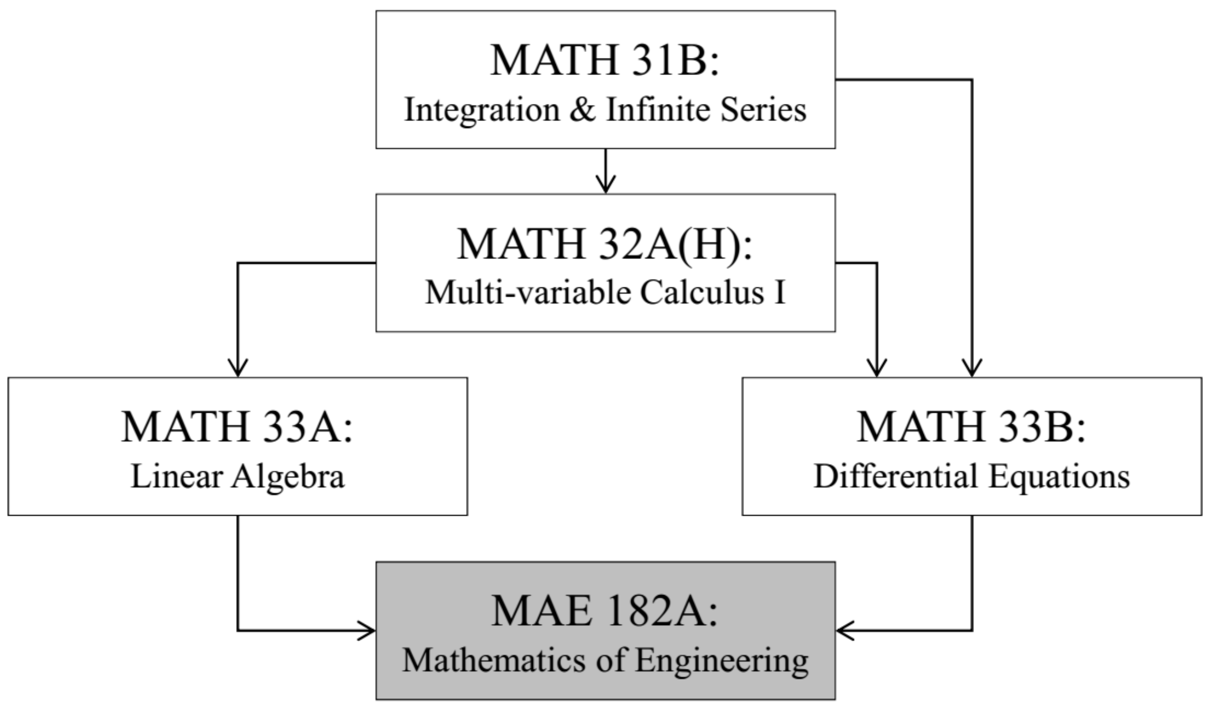
K = 20



K = 5

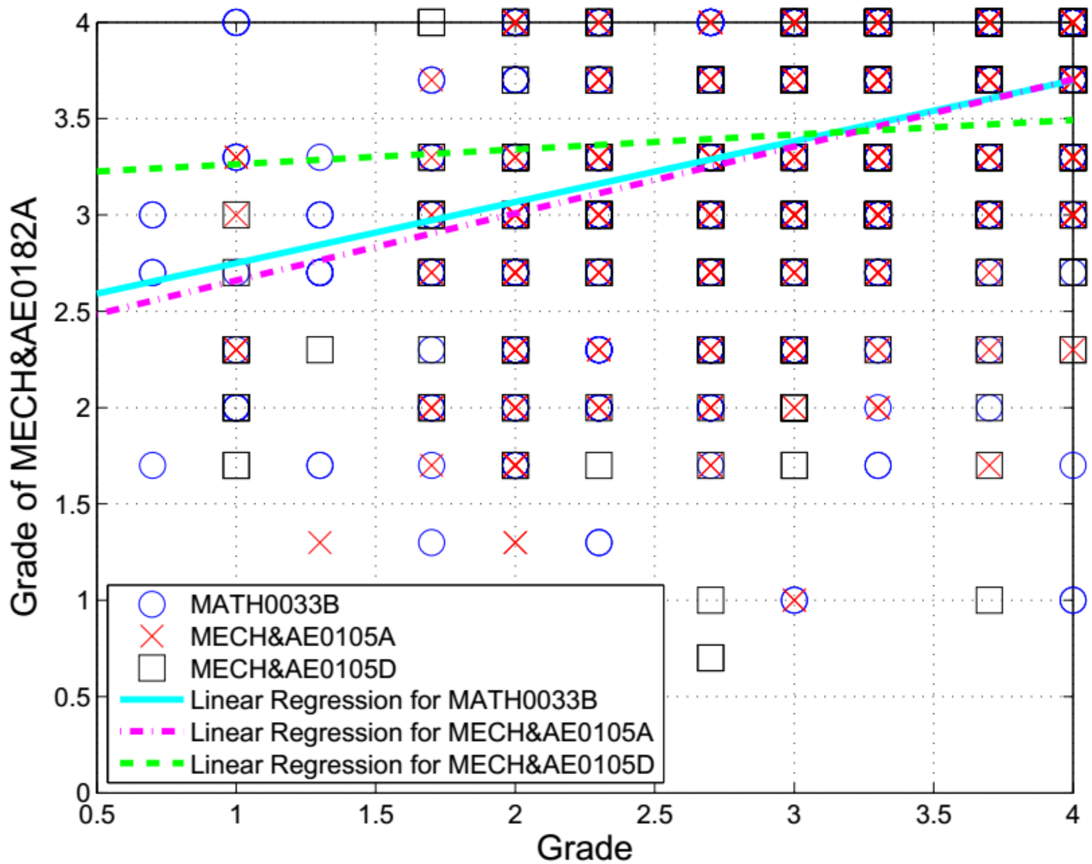
Correlated Courses: Case Study

- MAE 182A (Mathematics of Engineering)
  - Correlated courses according to prerequisites: MATH 31B, MATH 32A, MATH 33A, MATH 33B



Correlated Courses: Case Study

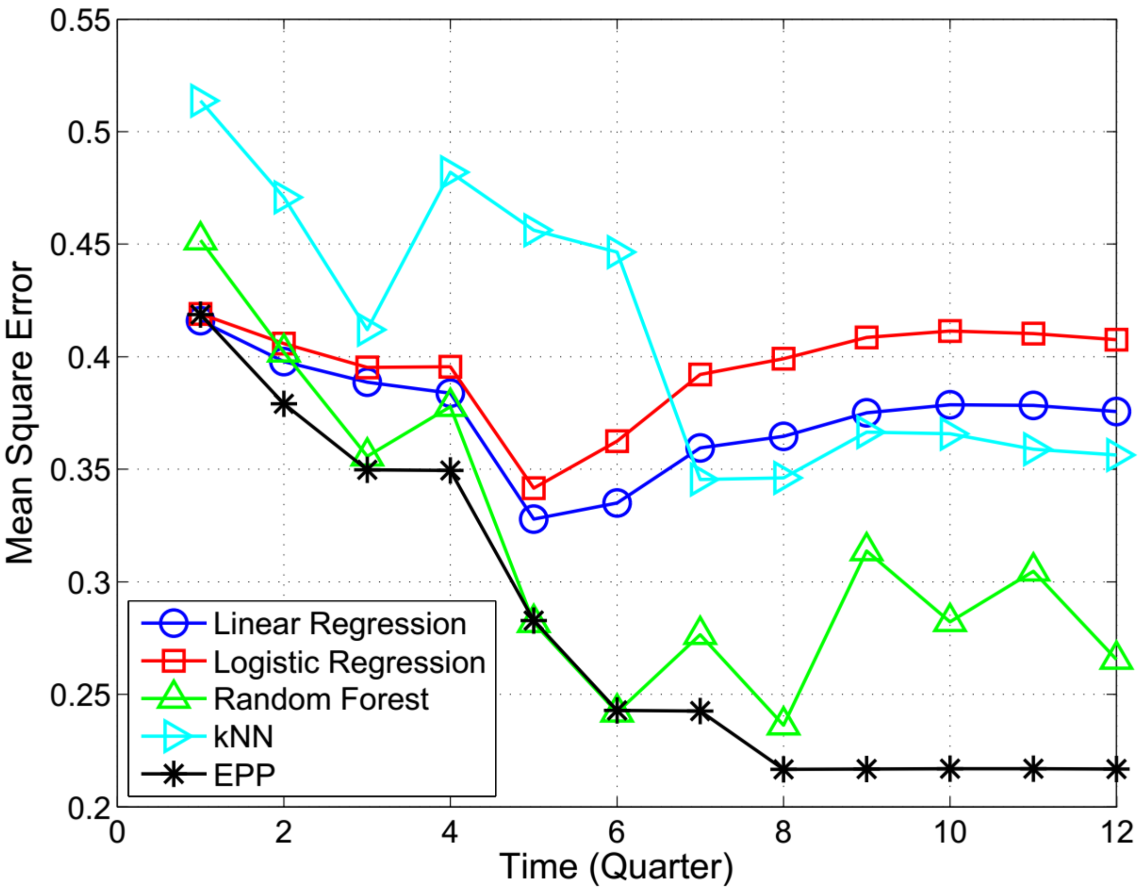
- MAE 182A (Mathematics of Engineering)
  - Our method discovers additional correlated courses: CHEM 20BH, EE 110L, MAE 102, MAE 105A, PHYS 1A



MAE 105A is correlated with MAE 182A
MAE 105D is not as correlated

Prediction Performance

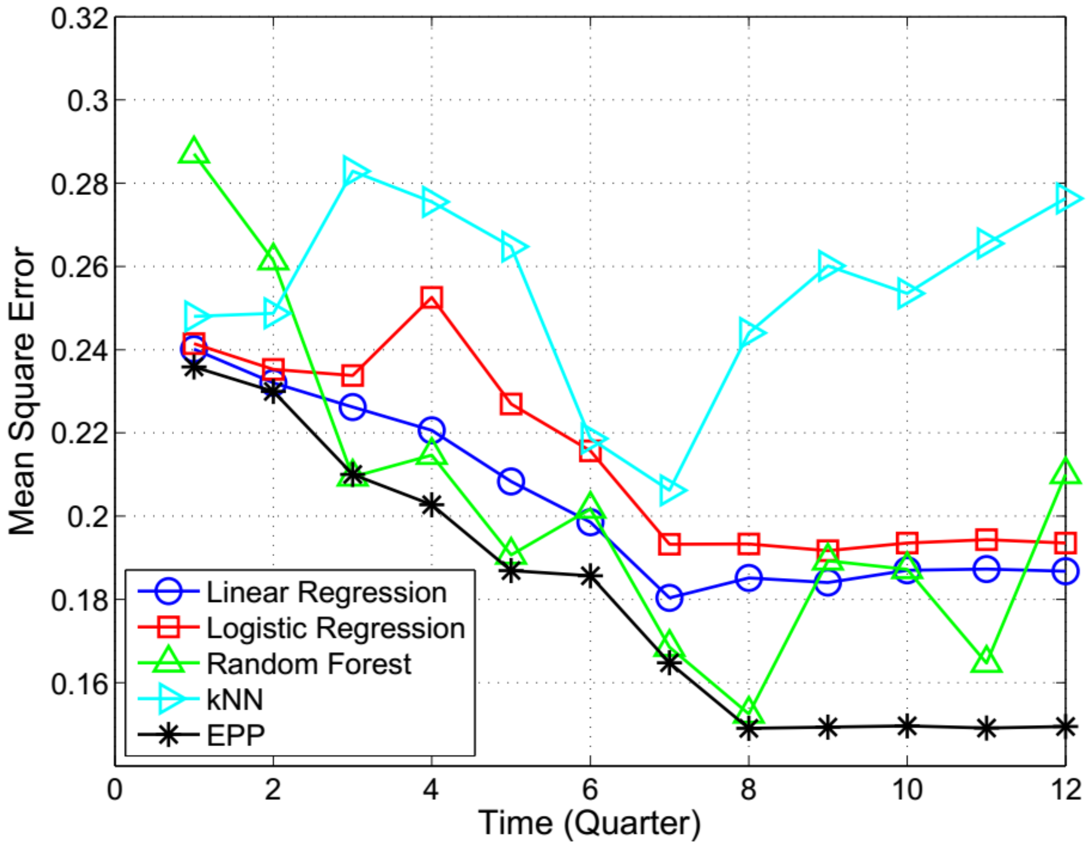
- Base vs Our Ensemble
  - Base predictors are implemented using linear regression, logistic regression, random forest, kNN



MAE 182A

Prediction Performance

- Base vs Our Ensemble
  - Base predictors are implemented using linear regression, logistic regression, random forest, kNN

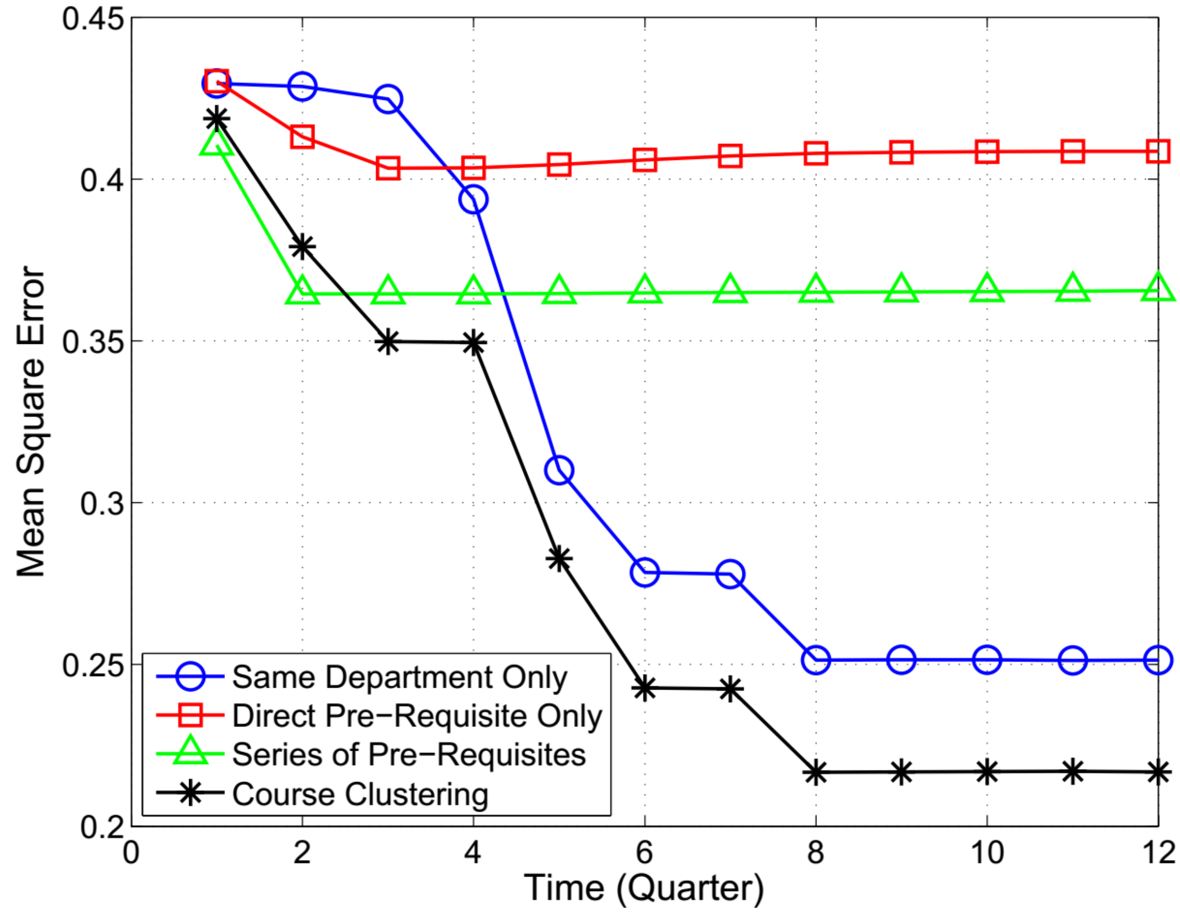


EE 110L

Prediction Performance

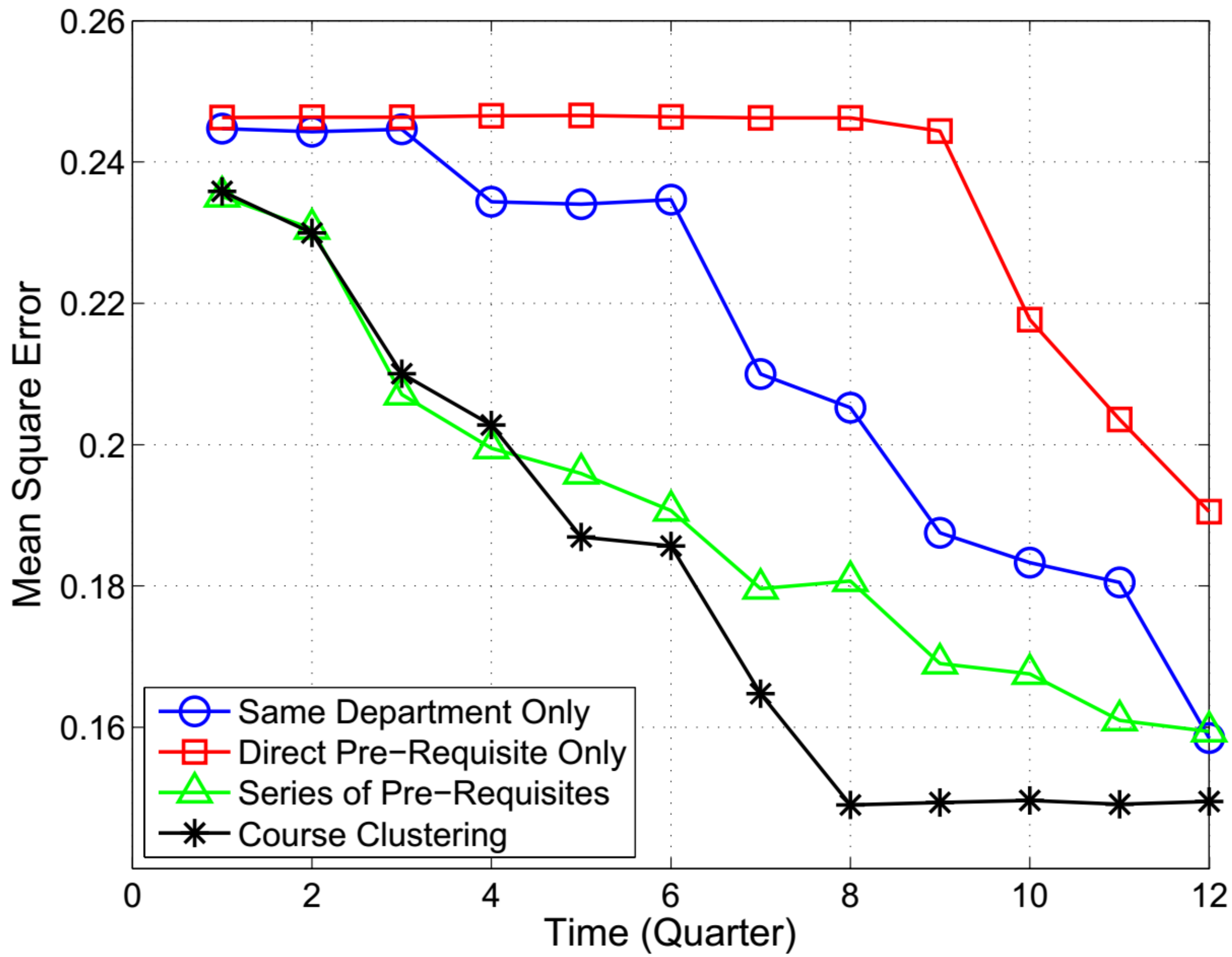
- Benchmarks using different input features
  - Same department only
    - Only courses offered by same department
  - Direct prerequisite only
  - Series of prerequisite
    - Include prerequisites of prerequisites

Prediction Performance



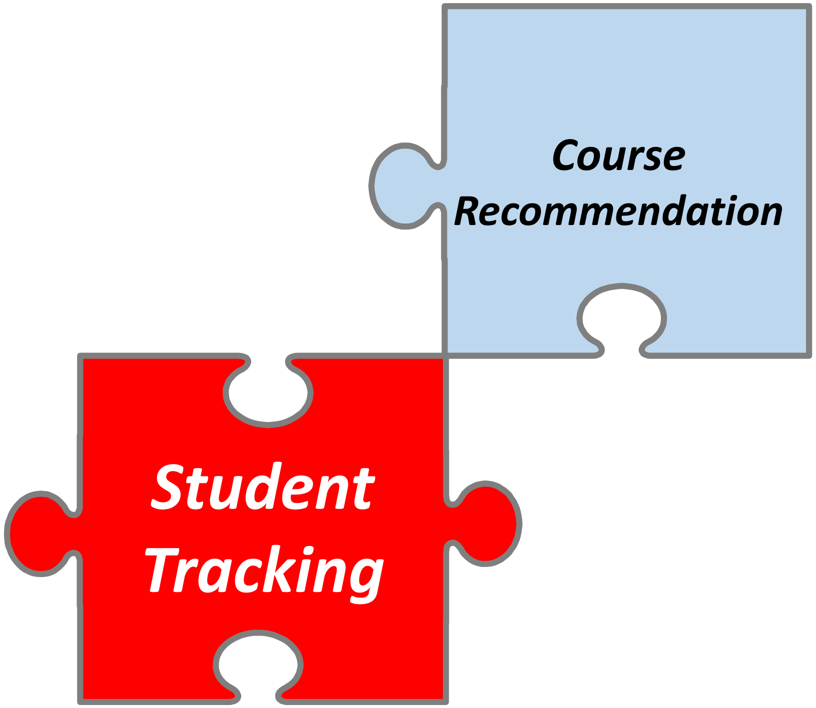
MAE 182A

Prediction Performance



EE 110L

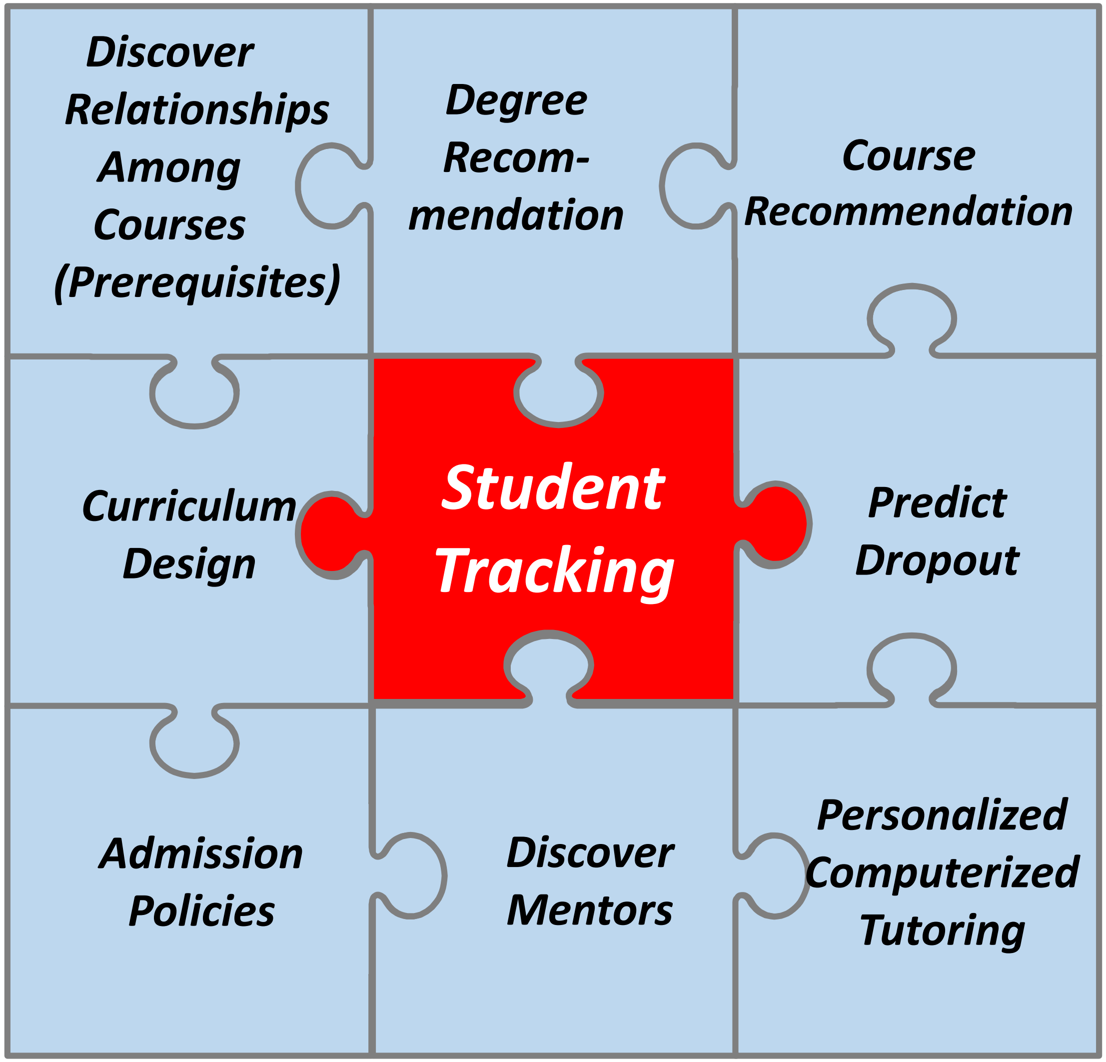
EPIE



W. Hoiles and M. van der Schaar, "Bounded Off-Policy Evaluation with Missing Data for Course Recommendation and Curriculum Design" *ICML, 2016.*

J. Xu, T. Xiang and M. van der Schaar, "Personalized Course Sequence Recommendations, " *IEEE Transactions on Signal Processing,* vol. 64, no. 20, pp. 5340-5352, Oct. 2016.

EPIE



http://medianetlab.ee.ucla.edu/EduAdvance