Active Learning: Efficient use of data labels

KDD, London, 22 Aug 2018

John-Mark Agosta, Olga Liakhovich, Robert Horton, Mario Inchiosa, Justin Ormont, Vanja Paunić, Siddarth Ramesh, Tomas Singliar, Ali-Kazim Zaidi, and Hang Zhang (Microsoft)

Active Learning makes efficient use of samples when labels are expensive

- Active Learning intersperses labelling of samples with incremental re-training. The currently trained model is used to select new samples to label. There are different search methods for selecting samples, depending on the data and model.
- Active Learning with just random search reduces to incremental learning. We compare learning curves for random search with active learning as a basis for comparison.

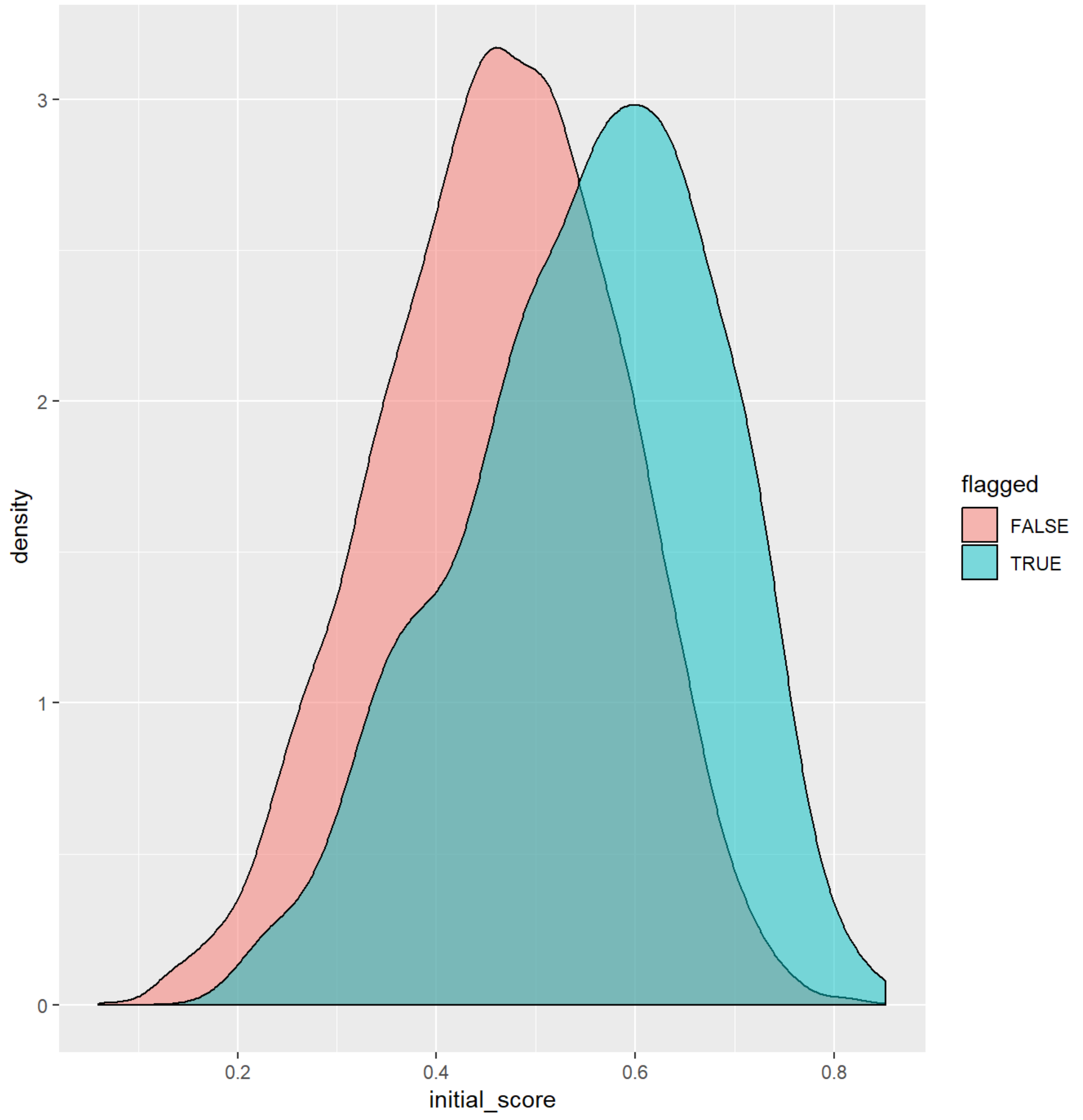
Algorithm Sketch

1. **Given an initial model, *M* and unlabeled set of samples *U:***
   1. **Using the current model *M* make class *c* likelihood predictions, *P(c | U) = M(U)*.**
   2. **Select a set to label *L* (possibly one) from *U,* based on *P(c | U).***
   3. **Update *M* with the training set *T' <- T + L***
2. **Repeat until model improvement / labeling cost < threshold**

Active Learning methods differ by the way they use *P(c | U)* to search over the set *U.*

uncertainty sampling

- The intuition is that unlabeled samples that the model predicts with greater uncertainty are more likely to be informative.
- This implies a strategy to select samples assigned the most uninformative probability by the current class *c* likelihood *P(c | U).*



Three uncertainty-sampling sample selection methods

Methods differ by how *P(c | U)* is used to select from *U.*

- Query Synthesis [Anguin, 1988]
- Selective Sampling [Atlas et al. 1989]
- Pool-based Active Learning [Lewis & Gale, SIGIR 1994] [Text classification - Lewis & Gale ICML 1994]

Query Synthesis

- Generate a query of where to look in the feature space *x* to select items to label. This applies with feature space representations where a “sample” could mean generating an *x ab initio* rather than selecting from an existing set *U.*
- For example *x* could be an chemical synthesis, or a synthetic image whose outcome is passed to the model learner.
- There’s a rough analog to the generative step in current DNN adversarial networks.

Selective Sampling

- Choose a region in feature space to focus on that is predicted to have the greatest uncertainty or information gain.
- This applies best when gaining new samples is passive or free, such as when selecting from streaming samples.
- Samples are selected sequentially from the stream.
- Unlike with *Query Synthesis,* samples are guaranteed to represent the actual distribution of the data, *P(U).*
- For example, generate image samples by aiming a camera at areas that need clarification.

Pool-based Sampling

- Choose greedily among the existing set of unlabeled samples *U* by an uncertainty measure applied to each element in the set.
- Batch sampling: one or more samples may be selected at each stage.
- When labelling costs vary among samples they may also be considered along with information gain.
- The examples in this tutorial will demonstrate Pool-based Sampling.

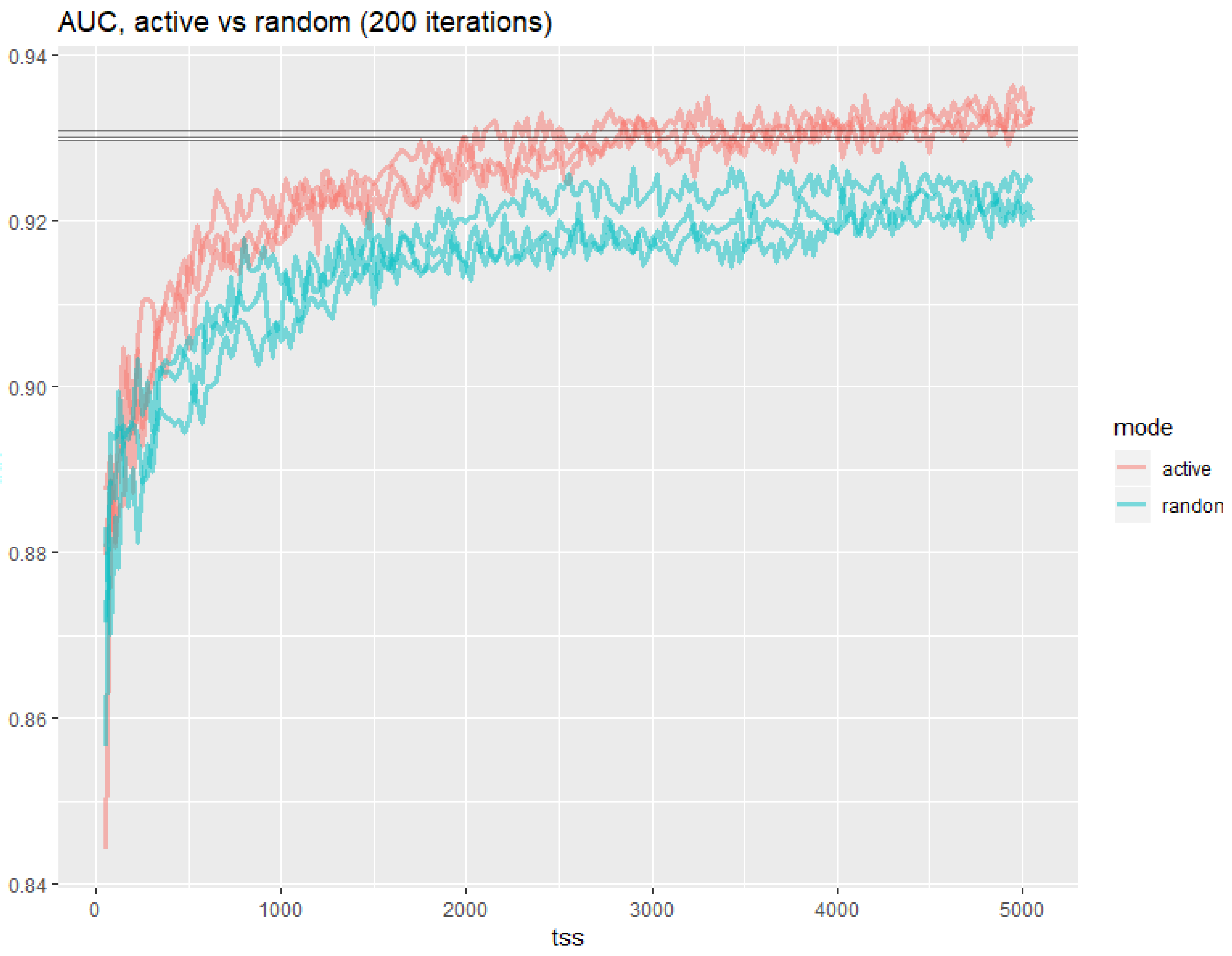
How Sampling Fails

- $P(c \mid u)$ often picks samples that are irrelevant, since there are areas of the sample space that are uncertain but do not help distinguish classes.
- Example: outliers may be highly uncertain but uninformative.
- An example is show here: [need picture]
- Better to consider the class likelihood $P(c \mid u, x)$ in the areas of feature space likely to distinguish known classes.
- Learners that generate margins for the class separators can find unlabeled samples both uncertain and that discriminate strongly between classes.

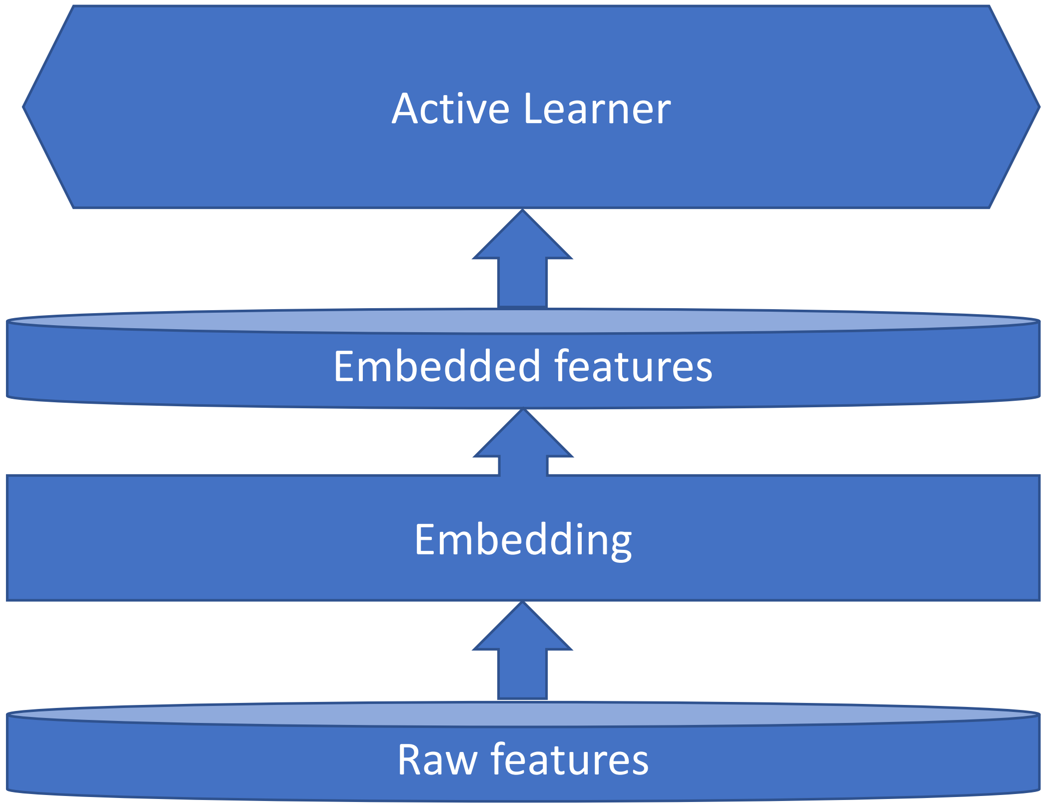
Working in the Version Space

- The space of all hypotheses is called the *version space*. Think all possible separators for a linear classifier.
- The version space is the dual to the feature space. Active Learning can be posed as maximizing the reduction in the version space by choice of samples.
- For instance, a sample that eliminates half the version space would best reduce model uncertainty.
- When model predictions are uncertain there are also Bayesian interpretations of the version space.

Learning curves show expected gains from Active Learning



How to exploit Transfer Learning



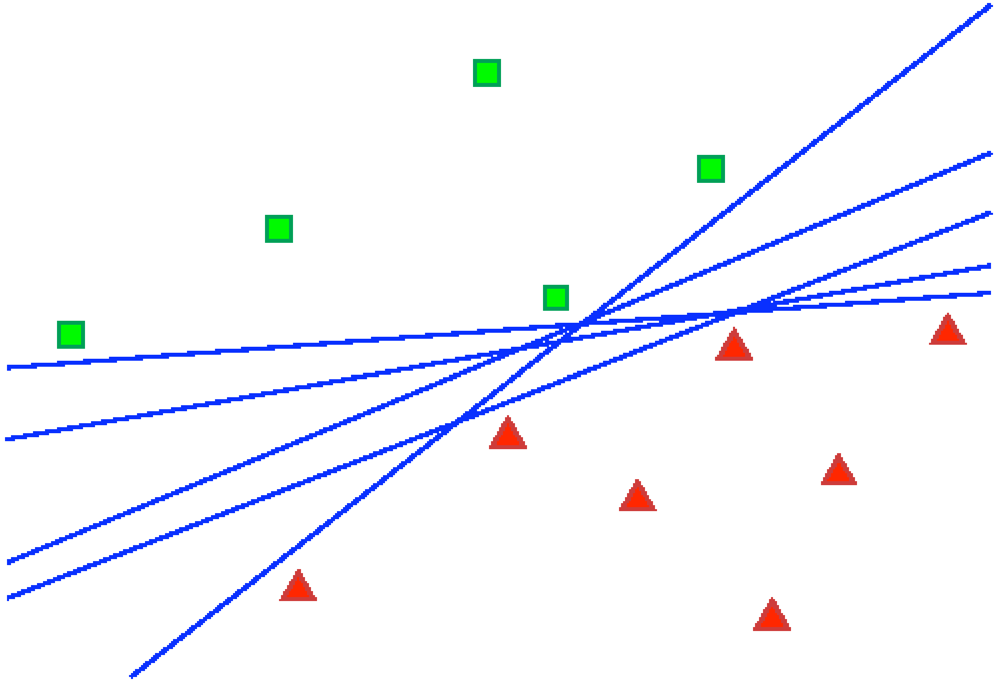
- Active Learning occurs on the embedded features
- The embedding transformation is pre-computed, just once.

Summary

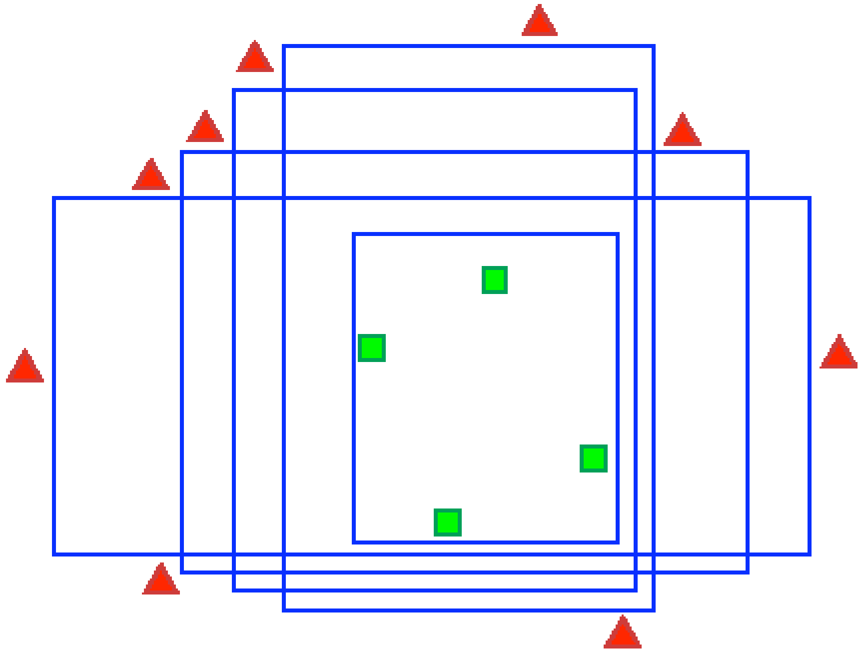
- Active Learning comes in many shapes and sizes
- Most any reasonable heuristic shows significant learning curve improvements
- But the theory is incomplete. My guess is that Wolpert's "no free lunch" theorem applies

Reference: Bart Settles (CMU) "Active Learning" (Morgan & Claypool, 2012).

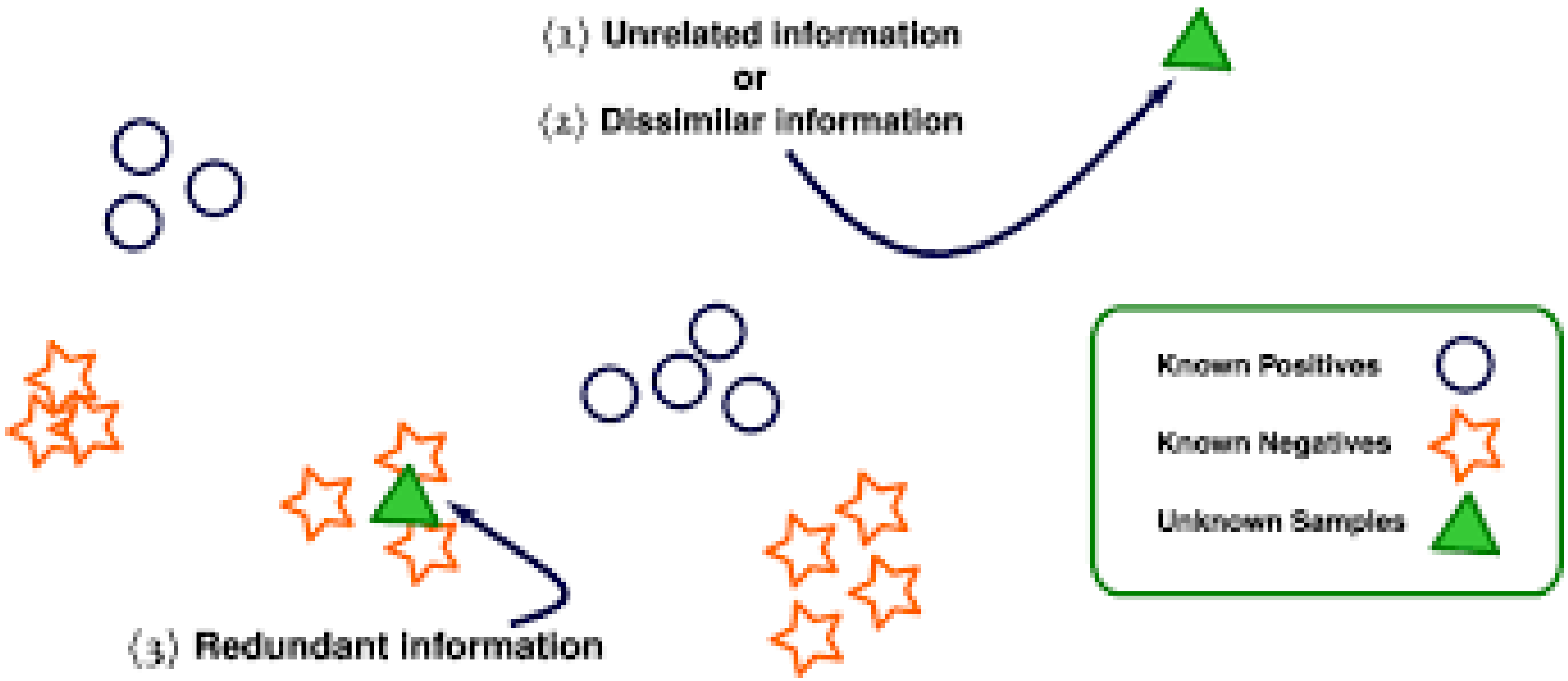
Version space



(a)



(b)



(1) Unrelated information
or
(2) Dissimilar information
(3) Redundant information
Known Positives
Known Negatives
Unknown Samples