SVM+ Prediction of Preterm Birth

Andrew A. Stirn
Columbia University
Computer Science Department
New York, NY, 10027
Email: andrew.stirn@columbia.edu

Joshua A. Rutta
Columbia University
Electrical Engineering Department
New York, NY 10027
Email: jar2317@columbia.edu

***Abstract*—Preterm birth (PTB) is the leading cause of neonatal mortality. Recent work has harnessed the power of machine learning to predict PTB at various stages of pregnancy. Naturally, as a pregnancy progresses the data that empowers prediction grows in dimensionality as more information becomes available with time. Consequently, PTB prediction in the earliest stages of pregnancy remains the most difficult for a classifier. To overcome this challenge, we propose using SVM+ and the Learning Using Privileged Information (LUPI) framework to incorporate future (privileged) data in the SVM+ optimization objective. We demonstrate that SVM+, which trains using past, current, and future data, will make better predictions off of past and current data than comparable models trained with only past and current data.**

I. Introduction

Preterm birth (PTB) is the leading cause of neonatal mortality and long-term disability. Alongside the obvious emotional aftermath of a PTB with an unfavorable outcome is a significant societal economic burden of \$26.2 billion (or \$51,600 per preterm infant born) for the United States in 2005 [1]. Thus, there exists significant impetus to develop methods to robustly predict PTB risk, both to identify at-risk women that require medical intervention and to rule out costly intervention for low-risk mothers.

II. Related Work

A. Preterm Birth Prediction

Recent work has applied supervised machine learning techniques to PTB prediction [2]. In particular, they focus on predicting spontaneous PTB (those that occur following the onset of spontaneous preterm labor, pre-labor rupture of the membranes, or cervical insufficiency), PTB for nulliparous (first pregnancy) women, and PTB of any kind. For each of these PTB data subsets, they train discriminative models to predict between a PTB and a full-term pregnancy at three temporal checkpoints throughout the pregnancy–those corresponding to major medical office visits. Thus, each model architecture is trained using three different subsets of data, each with varying dimensionality. Namely, a temporal instance of a model architecture is trained with currently and previously collected data. As the data becomes richer, the models naturally make better predictions. They use Logistic Regression and SVM with various kernels to show a marked improvement over previous work in sensitivity and specificity, achieving accuracy rates of 60%.

B. Learning Using Privileged Information

Given the temporally evolving nature of the data, we believe we can bolster the performance of these temporal model instantiations by training on both observable (past and present) data and unobservable (future) data while only predicting on the observable. In this manner, the unobservable future data can be considered privileged and thus we can apply the Learning Using Privileged Information (LUPI) paradigm [3]. This framework leverages the use of privileged information, and shows significant promise in achieving higher performance. SVM+ is a known method for implementing the LUPI paradigm [4], which can be formulated as a convex optimization problem (equation 1). Here, x^* is privileged information used only during training. Note that this formulation assumes $y_i \in \{-1, 1\}$.

$$\begin{aligned} \underset{w,b,w^*,d}{\text{minimize}} \; & \frac{1}{2}||w||_2^2 + \frac{\gamma}{2}||w^*||_2^2 + C\sum_{i=1}^{N}(w^* \cdot x_i^* + d) \\ \text{s.t.} \; & y_i(w \cdot x_i + b) \geq 1 - (w^* \cdot x_i^* + d), \; \forall \; i \in \{1, \dots, N\} \\ & w^* \cdot x_i + d \geq 0, \; \forall i \in \{1, \dots, N\} \end{aligned} \tag{1}$$

Predictions are made using only non-privileged information according to equation 2, which is also the prediction formula for regular SVM [5].

$$\hat{y}_i = \text{sign}(w \cdot x_i + b) \tag{2}$$

Comparing SVM+ (equation 1) with regular SVM (equation 3) is useful in understanding how privileged (future) data may produce a better classifier. Again, our SVM formulation assumes $y_i \in \{-1, 1\}$. Observe how SVM+ replaces the slack variables, u_i, with an affine operation on the privileged features. In essence, future data will govern the slack applied to each training example such that a better decision boundary might be learned. Outside of this intuition, SVM+ is known to have better convergence properties [3]. Specifically, if the error-correcting space (privileged data) is good, then the convergence (w.r.t. training set size) in the combined space could be on the order of $1/n$, as opposed to an algorithm operating only in the decision space (observable data), which converges on the order of $1/\sqrt{n}$, amounting to a difference of needing 320 training examples versus 100,000 training examples.

$$\begin{aligned} \underset{w,b,u}{\text{minimize}} \ & \frac{1}{2}||w||_2^2 + C\sum_{i=1}^{N} u_i \\ \text{s.t.} \ & y_i(w \cdot x_i + b) \geq 1 - u_i, \ \forall \ i \in \{1, \dots, N\} \\ & u_i \geq 0, \ \forall i \in \{1, \dots, N\} \end{aligned} \tag{3}$$

This LUPI property offers strong theoretical support for our application of SVM+ to construct a robust classifier that leverages future data at training time. The original work [2] showed for SVM-based models that prediction performance generally improves as more features become available. This improvement implies that the additional features are important for making better predictions. Therefore, we hypothesize that future data can be used as an error-correcting space to improve the convergence properties of the classifier. Additionally, we expect these gains to be amplified given the limited number of training examples discussed in section III.

III. DATA

Like [2] we will use data resulting from the "Preterm Prediction Study." This study was run by the National Institute of Child Health and Human Development (NICHD) Maternal Fetal Medicine Units (MFMU) Network between 1992 to 1994. In fact, we will use the same preprocessed dataset to ensure our results are comparable (see acknowledgments). The data was collected at six established checkpoints during the pregnancy. We present the temporal data emissions in an HMM-like diagram in figure 1. When a subject became pregnant at $t0$, prior information including demographic and previous pregnancy history was collected. The next four checkpoints ($t1 - t4$) were interleaving major and minor medical office visits. These visits produced a myriad of data including additional demographic data and a significant amount of biomedical data. By the final visit ($t4$), there are 328 total feature dimensions. The sixth time point ($t5$) was the actual delivery and included outcome results. Approximately 3000 women participated in the study. However, only 2327 of those subjects completed all four office visits. To apply the LUPI framework while avoiding data imputation, we consider only these 2327 subjects. Of these 2327 mothers, 285 suffered a PTB–leading to a large data set imbalance that we address in section V. Additionally, we only focus on predicting PTB of any kind as the data imbalance worsens when considering only nulliparous women.

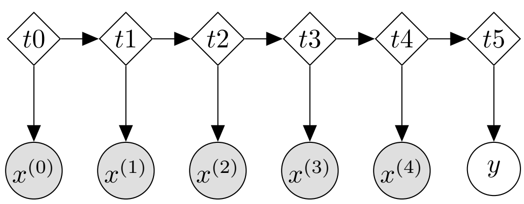

Fig. 1. Temporal Data Arrangement

A. LUPI Application

The LUPI application for the MFMU dataset is very straightforward. A model trained at time, t, will declare all data collected at or before t as the decision space (observable data upon which predictions are made). Similarly, a model built at time, t, will utilize future data as the error correcting space (unobservable data used only for training–not for prediction). This process is defined formally in set notation in equation 4. Note that we drop the (t) in all other formulas for notational simplicity. Also, when all data becomes observable, we replace the privileged feature space with a constant value of 1 for comparative purposes. We note however, that this replacement will inhibit SVM+ to optimize per-example slack variables. In this case, SVM+ can optimize global slack via w^* and d. This insufficiency is inconsequential since one would elect for regular SVM when there is no access to an error correcting space.

$$\begin{aligned} x(t) &= \{x^{(j)} \mid j \leq t\} \\ x^*(t) &= \begin{cases} \{x^{(j)} \mid j > t\} & , \ t \leq t4 \\ 1 & , \text{ o.w.} \end{cases} \end{aligned} \tag{4}$$

IV. METHODS

In addition to SVM+ (equation 1), we also consider Logistic Regression with l-2 regularization (equation 5 where $y_i \in \{0, 1\}$) and regular SVM (equation 3). We consider these alternative convex models both to measure any improvement with SVM+ and to confirm the results of the original work [2].

$$\underset{w,b}{\text{minimize}} \ \gamma||w||_2^2 + \sum_{i=1}^{N} \ln(1 + \exp(w \cdot x_i + b)) - y_i(w \cdot x_i + b) \tag{5}$$

For SVM and SVM+, we also consider Gaussian kernels. At training time, we replace x_i with $K_{:,i}$, the corresponding column of the Kernel matrix ($K \in \mathbf{S}_+^N$) where each entry is computed according to equation 6.

$$K_{i,j} = \phi(x_i, x_j) = \exp\left(-\frac{||x_i - x_j||_2^2}{2}\right) \tag{6}$$

To make a prediction for a new sample, x', we construct a kernelized feature vector according to equation 7, where $x_1 \dots x_N$ correspond to training examples (i.e. those that may form support vectors).

$$K' = \begin{bmatrix} \phi(x', x_1) \\ \vdots \\ \phi(x', x_N) \end{bmatrix} \in \mathbf{R}^N \tag{7}$$

We show the SVM dual problem in equation 8 where $X \in \mathbf{R}^{N \times \dim(x)}$ is a row-wise concatenation of our training features. Solving SVM or SVM+ in the dual space can offer two distinct advantages. First, for large N, training is more efficient because it works on the inner product space where the sample weights, α_i, are scalars rather than vectors. Second, it showcases the benefit of the well-known Kernel Trick.

Namely, only those α_i corresponding to examples near the decision boundary will be non-zero. The Representer Theorem states that our weights, w, in the primal problem can be represented as a linear combination of kernelized features (equation 9). With a sparse α, w is efficiently computed from just a small subset of x (or kernelized x).

$$\begin{aligned} \underset{\alpha}{\text{maximize}} \quad & 1^T\alpha - \frac{1}{2}\alpha^T(y \odot X)(y \odot X)^T\alpha \\ \text{s.t.} \quad & y^T\alpha = 0 \\ & 0 \preceq \alpha \preceq C \end{aligned} \tag{8}$$

However, since our training set is rather small, we train both SVM and SVM+ in their primal form. In fact, testing with the MFMU dataset showed only a marginal speedup when solving SVM with a Gaussian kernel in the dual with CVXPY [6].

$$w = \sum_{i=1}^{N} \alpha_i y_i x_i \tag{9}$$

Thus, in total, we employ the following five convex models (and associated kernels) solved in their primal forms for each of the five modeling time positions ($t0$-$t4$):

- Logistic Regression with Weight Regularization
- SVM with a Linear Kernel
- SVM with a Gaussian Kernel
- SVM+ with a Linear Kernel
- SVM+ with a Gaussian Kernel

V. Experiments

A. Addressing Data Imbalance

As mentioned, our dataset contains 2327 examples with only 285 positive elements (PTB occurrences). The dataset is limited in the total number of training examples when considering there are 328 feature dimensions by $t4$. Additionally, the dataset is severely imbalanced with many more negative training examples (normal births). To address the data imbalance and analyze performance variance, we employ Monte-Carlo cross-validation with randomized dataset balancing. In particular, we generated 50 randomized data folds. For each fold, we perform the following operations:

- Randomly balance the data set such that there is an equal number of positive and negative training examples.
- Randomly split the balanced fold into training (90%) and test (10%) sets.

B. Hyper-Parameter Selection

All three of our model families (Logistic Regression, SVM, and SVM+) have hyper-parameters that greatly affect model performance. Therefore, for each model family, we declared a finite hyper-parameter search space. To seek optimal hyper-parameters, we trained each model family and associated kernel options for all hyper-parameters on the 50 randomized training sets and evaluated performance on the 50 corresponding test sets. This process ensured that each combination of model, kernel, and hyper-parameter(s) saw the same 50 data folds such that mean performance and its variance can be assessed fairly.

For logistic regression, γ controls the strength of l-2 weight normalization (equation 5). One may choose to increase γ in the presence of overfitting. For logistic regression, we consider the following γ values:

$$\gamma \in \{\text{1e+00, 1e+01, 1e+02, 1e+03}\} \tag{10}$$

For SVM, C controls the influence of slack variables (equation 3). Lowering C decreases the penalty for increasing slack. A low C will encourage a small a margin that best fits the training data. Increasing C will allow more test error but find a wider margin that perhaps generalizes better to unseen data. For SVM, we consider the following C values:

$$C \in \{\text{1e-04, 1e-03, 1e-02, 1e-01, 1e+00, 1e+01}\} \tag{11}$$

SVM+ has both γ and C hyper-parameters (equation 1). Here, C again controls the margin width similarly to regular SVM but instead operates on the slack generated by the affine transformation on the privileged data. Whereas γ restricts the capacity of the error correcting space by penalizing large $||w^*||_2^2$. In essence, larger γ restricts the variance of the utilized slack in SVM+. For SVM+, we consider every combination of the following C and γ values:

$$\begin{aligned} C &\in \{\text{1e-06, 2e-06, 5e-06, 1e-05}\} \\ \gamma &\in \{\text{1e+00, 1e+01, 1e+02, 1e+03}\} \end{aligned} \tag{12}$$

For all model families, these hyper-parameter sets were defined as a result of preliminary tuning work on randomized individual data folds. They represent a range of parameters that seemed to produce good performance at each time step for all model and kernel combinations. In total, we trained 48 model configurations across the five model-kernel pairs and their respective hyper-parameters.

VI. Results

In this section, we present the results generated from our experimental methods described in section VI. To assess performance, we consider recall, precision, and F1 scores defined respectively in equations 13, 14, 15.

$$\text{Recall} = \frac{\text{true positives}}{\text{true positives} + \text{false negatives}} \tag{13}$$

$$\text{Precision} = \frac{\text{true positives}}{\text{true positives} + \text{false positives}} \tag{14}$$

$$\text{F1} = \frac{2}{\frac{1}{\text{Recall}} + \frac{1}{\text{Precision}}} \tag{15}$$

Recall from section V that we trained 48 model configurations at each time step for all 50 randomized data folds. To condense our report, we only show the top-performance results (with respect to mean F1 score) for each of the five model-kernel combinations at each time point, $t0 - t4$. These results appear chronologically in tables I, II, III, IV, and V. For full results including the F1 score variance, we refer the interested reader to the appendix (section VIII).

Algorithm	Log Reg	SVM	SVM	SVM+	SVM+
Kernel	Lin	Lin	Gauss	Lin	Gauss
γ	1e+01	–	–	1e+02	1e+03
C	–	1e-03	1e-03	1e-06	1e-06
Recall	0.310	0.532	0.728	0.983	**0.994**
Precision	0.665	0.593	0.557	**0.593**	0.564
F1	0.418	0.558	0.629	**0.665**	**0.665**

TABLE I
EACH MODEL'S TOP PERFORMANCE AT $t0$

Algorithm	Log Reg	SVM	SVM	SVM+	SVM+
Kernel	Lin	Lin	Gauss	Lin	Gauss
γ	1e+00	–	–	1e+03	1e+03
C	–	1e-03	1e-02	1e-06	1e-06
Recall	0.486	0.561	**0.898**	0.849	0.820
Precision	0.580	**0.600**	0.525	0.583	0.505
F1	0.525	0.576	0.0662	0.666	**0.667**

TABLE II
EACH MODEL'S TOP PERFORMANCE AT $t1$

Algorithm	Log Reg	SVM	SVM	SVM+	SVM+
Kernel	Lin	Lin	Gauss	Lin	Gauss
γ	1e+01	–	–	1e+03	1e+00
C	–	1e-03	1e-01	1e-06	1e-06
Recall	0.470	0.563	0.911	0.879	**0.931**
Precision	0.606	**0.607**	0.530	0.593	0.523
F1	0.525	0.580	**0.669**	0.666	**0.669**

TABLE III
EACH MODEL'S TOP PERFORMANCE AT $t2$

Algorithm	Log Reg	SVM	SVM	SVM+	SVM+
Kernel	Lin	Lin	Gauss	Lin	Gauss
γ	1e+02	–	–	1e+02	1e+01
C	–	1e-01	1e-02	1e-06	2e-06
Recall	0.486	0.556	**0.934**	0.810	0.020
Precision	**0.653**	0.652	0.524	0.629	0.071
F1	0.553	0.596	0.671	0.636	**0.674**

TABLE IV
EACH MODEL'S TOP PERFORMANCE AT $t3$

Algorithm	Log Reg	SVM	SVM	SVM+	SVM+
Kernel	Lin	Lin	Gauss	Lin	Gauss
γ	1e+01	–	–	1e+00	1e+03
C	–	1e-01	1e+01	2e-06	1e-06
Recall	0.525	0.552	**0.937**	0.597	**0.937**
Precision	0.604	0.646	0.525	**0.658**	0.525
F1	0.559	0.591	**0.673**	0.622	**0.673**

TABLE V
EACH MODEL'S TOP PERFORMANCE AT $t4$

The original work [2] only trained models for $t0$, $t1$, and $t3$. Our overlapping models performed comparably when trained on the full dataset (all types of PTB). Our results for Logistic Regression and SVM with a linear kernel perform slightly worse than their reported performances. Conversely, our SVM with a Gaussian kernel performs better. Thus, we feel confident in both their numbers and ours.

As hypothesized, SVM+ generally outperforms the other models. In fact, SVM+ with the Gaussian kernel either comes in first or ties for first (with respect to F1) at all time steps.

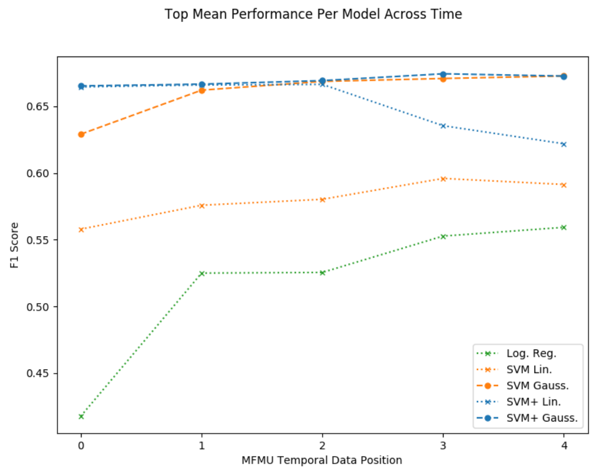

Fig. 2. Mean F1 Performance Corresponding to the Best Hyper-Parameters Over Time for Each of the Five Model-Kernel Combinations

SVM+ with a linear kernel also does quite well beating out all Logistic Regression and SVM variants until $t2$. Thus, it seems that the privileged future data constitutes a good error correcting space for the PTB prediction problem.

We also examine the mean F1 performance trajectory across time for the five model-kernel combinations. We plot the top F1 scores for the model-kernel performances listed in tables I through V in figure 2, from which a few insights can be obtained. First, notice that all models without access to the privileged (future) error correcting data space naturally improve with time as the data becomes richer. Conversely, SVM+ with a Gaussian Kernel is at peak performance throughout, seeing almost negligible performance gain through $t3$ and a negligible drop at $t4$.

We also would like to point out that SVM+ with a linear kernel oddly begins to degrade after $t3$. We cannot make any concrete claims regarding this phenomena, however, we will offer an explanation for $t4$ and apply parts of it inductively to $t3$. At $t4$ there is no privileged information and we therefore passed in scalar 1's for each training example. Regular SVM has control over N u_i's resulting in potentially unique slack for each training example. Passing in 1's for all x_i^* to SVM+ means that the N slack variables are controlled only by $w*$ and d. Thus, at $t4$, SVM+ is restricted to applying the same slack to all training examples: $w^* + d$. In essence, SVM+ loses its advantage as the privileged data shrinks in dimensions. We now inductively apply this argument to $t3$. At $t2$, $\dim(x_i^*) = 111$. At $t3$, $\dim(x_i^*) = 12$. Recall that in total there are 328 feature dimensions. Therefore, by $t3$, the privileged space has lost most of its dimensionality inhibiting the LUPI advantage. We suspect SVM+ with a Gaussian kernel gets around this problem by operating in higher dimensional feature space. Conversely, SVM+ with a linear kernel loses its ability to construct meaningful slack for each example and lacks a powerful kernel to fall back on.

VII. Conclusion

In summary, we considered the acutely socially and economically motivated problem of Preterm Birth (PTB) prediction. We treat the problem as a classification problem and utilized the same temporally evolving MFMU dataset discussed in the original work [2] on this problem. We both confirm their results and expand on their work by applying the Learning Using Privileged Information (LUPI) framework with SVM+ to show a demonstrable improvement in prediction performance. In particular, we use future data as an error correcting space during model training to bolster predictions that use only current and past data. Our performance gains are expected and in line with the LUPI theory presented in [3].

A. Future Work

It should be noted that an additional PTB dataset has been collected that is both larger and richer. We hypothesize that our results and the results of [2] are restricted by dataset size and imbalance. When this new data becomes available and is preprocessed, we suspect better performance numbers will be immediately attained.

To improve results on the current MFMU data and prepare for potential class imbalance in the new data, we suggest modifying the convex objectives to have weighted penalties corresponding to class populations. This technique is very simple to implement and avoids the need to throw away large swaths of data to balance the positive and negative examples before training.

Acknowledgment

We would like to thank Ansaf Salleb-Aouissi, PhD., for her guidance. In addition to sharing the data, her original work on this problem [2] using SVM and Logistic Regression inspired her to encourage our usage of SVM+ and LUPI for PTB prediction. This work is an original implementation of her idea.

Statement of Team Member Contributions

To preface, Joshua is pursuing his Master's degree in EE at Columbia after completing his undergraduate degree. Andrew has returned to school after nine years of professional software development experience to pursue a Master's degree in CS and hopefully a PhD. In this light, Andrew does have a higher level of software competency such that we feel it would unfair to discuss contribution levels without acknowledging the experience gap.

We began our project by first implementing all five model-kernel combinations discussed in section IV. Andrew built the SVM (primal and dual) and SVM+ models as well as the Gaussian kernel function. Joshua built the original Logistic Regression implementation and integrated it with Andrew's Gaussian kernel. However, Logistic Regression with a Gaussian Kernel had convergence issues with the CVXPY toolbox and thus was discarded. Joshua also attempted a different kernel integration for SVM.

We both tested our respective implementations against a common set of unit tests. For those tests, Andrew generated two simulated datasets in $\Re^2$: one that is linearly separable and one that is separable with an RBF. Both parties used these tests to visually inspect and confirm the performance of their models before we began work on the MFMU data. Joshua helped refine the unit test plots for the Gaussian kernel boundaries.

We both met with Professor Ansaf Salleb-Aouissi to go over the data. We then strategized how to most efficiently parse the data such that LUPI could be applied. Joshua contributed the winning idea on how to best parse the data into their corresponding privileged and non-privileged components for each time-step. Andrew implemented that idea and the rest of the associated software infrastructure including: data loading, data folding, experiment looping, result saving, and latex table exporting.

Reports were made in tandem by Joshua in PowerPoint and by Andrew in Latex. It was decided to settle on a Latex presentation. Finally, the report was a joint effort where we leveraged work from our proposal and presentation.

References

[1] R. Behrman and A. Butler, *Preterm Birth: Causes, Consequences, and Prevention*. Institute of Medicine Committee on Understanding Premature Birth and Assuring Healthy Outcomes, 2007. [Online]. Available: https://www.ncbi.nlm.nih.gov/books/NBK11362/

[2] I. Vovsha, A. Salleb-Aouissi, A. Raja, T. Koch, A. Rybchuk, A. Radeva, A. Rajan, Y. Huang, H. Diab, A. Tomar, and R. J. Wapner, "Using kernel methods and model selection for prediction of preterm birth," *CoRR*, vol. abs/1607.07959, 2016. [Online]. Available: http://arxiv.org/abs/1607.07959

[3] D. Pechyony and V. Vapnik, "On the theory of learning with privileged information," in *NIPS*. Curran Associates, Inc., 2010, pp. 1894–1902.

[4] V. Vapnik, "Learning hidden information: SVM+," in *GrC*. IEEE, 2006, p. 22.

[5] C. Cortes and V. Vapnik, "Support-vector networks," *Mach. Learn.*, vol. 20, no. 3, pp. 273–297, Sep. 1995. [Online]. Available: https://doi.org/10.1023/A:1022627411411

[6] S. Boyd and L. Vandenberghe, *Convex Optimization*. New York, NY, USA: Cambridge University Press, 2004.

VIII. Appendix

A. T0 Complete Results

γ = 1e+00	γ = 1e+01	γ = 1e+02	γ = 1e+03
0.416 ± 0.090	**0.418** ± 0.082	0.394 ± 0.092	0.207 ± 0.084

TABLE VI
LOGISTIC REGRESSION AT T0: F1 SCORE (MEAN ± STD.)

γ = 1e+00	γ = 1e+01	γ = 1e+02	γ = 1e+03
0.312 / 0.650	0.310 / 0.665	0.279 / 0.703	0.123 / 0.806

TABLE VII
LOGISTIC REGRESSION AT T0: RECALL (MEAN) / PRECISION (MEAN)

C = 1e-04	C = 1e-03	C = 1e-02	C = 1e-01	C = 1e+00	C = 1e+01
0.502 ± 0.097	**0.558** ± 0.080	0.491 ± 0.084	0.488 ± 0.105	0.507 ± 0.085	0.503 ± 0.080

TABLE VIII
SVM WITH LINEAR KERNEL AT T0: F1 SCORE (MEAN ± STD.)

C = 1e-04	C = 1e-03	C = 1e-02	C = 1e-01	C = 1e+00	C = 1e+01
0.430 / 0.629	0.532 / 0.593	0.409 / 0.632	0.408 / 0.642	0.443 / 0.606	0.443 / 0.592

TABLE IX
SVM WITH LINEAR KERNEL AT T0: RECALL (MEAN) / PRECISION (MEAN)

C = 1e-04	C = 1e-03	C = 1e-02	C = 1e-01	C = 1e+00	C = 1e+01
0.619 ± 0.062	**0.629** ± 0.057	0.620 ± 0.053	0.572 ± 0.064	0.542 ± 0.076	0.525 ± 0.072

TABLE X
SVM WITH GAUSSIAN KERNEL AT T0: F1 SCORE (MEAN ± STD.)

C = 1e-04	C = 1e-03	C = 1e-02	C = 1e-01	C = 1e+00	C = 1e+01
0.699 / 0.558	0.728 / 0.557	0.706 / 0.556	0.581 / 0.568	0.521 / 0.572	0.505 / 0.555

TABLE XI
SVM WITH GAUSSIAN KERNEL AT T0: RECALL (MEAN) / PRECISION (MEAN)

	γ = 1e+00	γ = 1e+01	γ = 1e+02	γ = 1e+03
C = 1e-06	0.562 ± 0.074	0.625 ± 0.077	**0.665** ± 0.073	0.663 ± 0.069
C = 2e-06	0.560 ± 0.058	0.580 ± 0.072	0.658 ± 0.073	0.661 ± 0.073
C = 5e-06	0.554 ± 0.016	0.567 ± 0.023	0.617 ± 0.063	0.649 ± 0.074
C = 1e-05	0.561 ± 0.025	0.567 ± 0.032	0.582 ± 0.033	0.606 ± 0.069

TABLE XII
SVM+ WITH LINEAR KERNEL AT T0: F1 SCORE (MEAN ± STD.)

	γ = 1e+00	γ = 1e+01	γ = 1e+02	γ = 1e+03
C = 1e-06	0.544 / 0.587	0.754 / 0.591	0.983 / 0.593	0.948 / 0.594
C = 2e-06	0.539 / 0.552	0.583 / 0.587	0.946 / 0.591	0.978 / 0.593
C = 5e-06	0.526 / 0.504	0.550 / 0.506	0.752 / 0.540	0.926 / 0.588
C = 1e-05	0.538 / 0.502	0.549 / 0.506	0.586 / 0.507	0.677 / 0.569

TABLE XIII
SVM+ WITH LINEAR KERNEL AT T0: RECALL (MEAN) / PRECISION (MEAN)

	γ = 1e+00	γ = 1e+01	γ = 1e+02	γ = 1e+03
C = 1e-06	0.597 ± 0.061	0.619 ± 0.058	0.643 ± 0.064	**0.665** ± 0.065
C = 2e-06	0.580 ± 0.063	0.616 ± 0.060	0.628 ± 0.060	0.657 ± 0.059
C = 5e-06	0.575 ± 0.045	0.597 ± 0.056	0.617 ± 0.066	0.632 ± 0.066
C = 1e-05	0.563 ± 0.007	0.576 ± 0.034	0.611 ± 0.058	0.617 ± 0.063

TABLE XIV
SVM+ WITH GAUSSIAN KERNEL AT T0: F1 SCORE (MEAN ± STD.)

	γ = 1e+00	γ = 1e+01	γ = 1e+02	γ = 1e+03
C = 1e-06	0.641 / 0.562	0.692 / 0.567	0.531 / 0.567	0.994 / 0.564
C = 2e-06	0.599 / 0.562	0.677 / 0.568	0.730 / 0.564	0.909 / 0.569
C = 5e-06	0.588 / 0.512	0.637 / 0.553	0.690 / 0.561	0.737 / 0.564
C = 1e-05	0.568 / 0.500	0.588 / 0.519	0.671 / 0.556	0.690 / 0.562

TABLE XV
SVM+ WITH GAUSSIAN KERNEL AT T0: RECALL (MEAN) / PRECISION (MEAN)

B. T1 Complete Results

γ = 1e+00	γ = 1e+01	γ = 1e+02	γ = 1e+03
0.525 $\pm$ 0.069	0.520 $\pm$ 0.076	0.508 $\pm$ 0.090	0.457 $\pm$ 0.086

TABLE XVI
LOGISTIC REGRESSION AT T1: F1 SCORE (MEAN $\pm$ STD.)

γ = 1e+00	γ = 1e+01	γ = 1e+02	γ = 1e+03
0.486 / 0.580	0.469 / 0.598	0.428 / 0.641	0.339 / 0.729

TABLE XVII
LOGISTIC REGRESSION AT T1: RECALL (MEAN) / PRECISION (MEAN)

C = 1e-04	C = 1e-03	C = 1e-02	C = 1e-01	C = 1e+00	C = 1e+01
0.518 $\pm$ 0.088	**0.576** $\pm$ 0.075	0.523 $\pm$ 0.084	0.566 $\pm$ 0.078	0.564 $\pm$ 0.083	0.544 $\pm$ 0.069

TABLE XVIII
SVM WITH LINEAR KERNEL AT T1: F1 SCORE (MEAN $\pm$ STD.)

C = 1e-04	C = 1e-03	C = 1e-02	C = 1e-01	C = 1e+00	C = 1e+01
0.441 / 0.654	0.561 / 0.600	0.448 / 0.650	0.529 / 0.618	0.546 / 0.590	0.537 / 0.558

TABLE XIX
SVM WITH LINEAR KERNEL AT T1: RECALL (MEAN) / PRECISION (MEAN)

C = 1e-04	C = 1e-03	C = 1e-02	C = 1e-01	C = 1e+00	C = 1e+01
0.599 $\pm$ 0.194	0.641 $\pm$ 0.091	**0.662** $\pm$ 0.028	0.660 $\pm$ 0.037	0.655 $\pm$ 0.037	0.656 $\pm$ 0.038

TABLE XX
SVM WITH GAUSSIAN KERNEL AT T1: F1 SCORE (MEAN $\pm$ STD.)

C = 1e-04	C = 1e-03	C = 1e-02	C = 1e-01	C = 1e+00	C = 1e+01
0.450 / 0.501	0.850 / 0.529	0.898 / 0.525	0.891 / 0.530	0.869 / 0.526	0.866 / 0.528

TABLE XXI
SVM WITH GAUSSIAN KERNEL AT T1: RECALL (MEAN) / PRECISION (MEAN)

	γ = 1e+00	γ = 1e+01	γ = 1e+02	γ = 1e+03
C = 1e-06	0.577 $\pm$ 0.076	0.580 $\pm$ 0.077	0.632 $\pm$ 0.072	**0.666** $\pm$ 0.059
C = 2e-06	0.583 $\pm$ 0.075	0.578 $\pm$ 0.075	0.628 $\pm$ 0.070	0.637 $\pm$ 0.073
C = 5e-06	0.588 $\pm$ 0.086	0.580 $\pm$ 0.102	0.592 $\pm$ 0.073	0.603 $\pm$ 0.071
C = 1e-05	0.583 $\pm$ 0.004	0.581 $\pm$ 0.070	0.579 $\pm$ 0.090	0.587 $\pm$ 0.076

TABLE XXII
SVM+ WITH LINEAR KERNEL AT T1: F1 SCORE (MEAN $\pm$ STD.)

	γ = 1e+00	γ = 1e+01	γ = 1e+02	γ = 1e+03
C = 1e-06	0.562 / 0.600	0.588 / 0.614	0.826 / 0.600	0.849 / 0.583
C = 2e-06	0.562 / 0.582	0.573 / 0.592	0.786 / 0.602	0.782 / 0.609
C = 5e-06	0.583 / 0.530	0.566 / 0.535	0.606 / 0.588	0.728 / 0.591
C = 1e-05	0.588 / 0.504	0.562 / 0.542	0.574 / 0.560	0.602 / 0.581

TABLE XXIII
SVM+ WITH LINEAR KERNEL AT T1: RECALL (MEAN) / PRECISION (MEAN)

	γ = 1e+00	γ = 1e+01	γ = 1e+02	γ = 1e+03
C = 1e-06	0.665 $\pm$ 0.025	0.379 $\pm$ 0.232	**0.667** $\pm$ 0.166	**0.667** $\pm$ 0.154
C = 2e-06	0.404 $\pm$ 0.308	0.591 $\pm$ 0.187	0.628 $\pm$ 0.031	**0.667** $\pm$ 0.033
C = 5e-06	0.602 $\pm$ 0.000	0.661 $\pm$ 0.153	0.650 $\pm$ 0.080	**0.667** $\pm$ 0.088
C = 1e-05	0.605 $\pm$ 0.000	0.661 $\pm$ 0.000	0.656 $\pm$ 0.000	**0.667** $\pm$ 0.000

TABLE XXIV
SVM+ WITH GAUSSIAN KERNEL AT T1: F1 SCORE (MEAN $\pm$ STD.)

	γ = 1e+00	γ = 1e+01	γ = 1e+02	γ = 1e+03
C = 1e-06	0.908 / 0.525	0.359 / 0.465	0.600 / 0.494	0.820 / 0.505
C = 2e-06	0.474 / 0.455	0.684 / 0.517	0.621 / 0.527	0.180 / 0.526
C = 5e-06	0.817 / 0.500	0.888 / 0.474	0.540 / 0.520	0.180 / 0.518
C = 1e-05	0.800 / 0.500	0.892 / 0.500	0.926 / 0.500	0.180 / 0.500

TABLE XXV
SVM+ WITH GAUSSIAN KERNEL AT T1: RECALL (MEAN) / PRECISION (MEAN)

C. T2 Complete Results

γ = 1e+00	γ = 1e+01	γ = 1e+02	γ = 1e+03
0.522 ± 0.069	**0.525** ± 0.077	0.507 ± 0.091	0.457 ± 0.093

TABLE XXVI
LOGISTIC REGRESSION AT T2: F1 SCORE (MEAN ± STD.)

γ = 1e+00	γ = 1e+01	γ = 1e+02	γ = 1e+03
0.481 / 0.578	0.470 / 0.606	0.427 / 0.644	0.343 / 0.715

TABLE XXVII
LOGISTIC REGRESSION AT T2: RECALL (MEAN) / PRECISION (MEAN)

C = 1e-04	C = 1e-03	C = 1e-02	C = 1e-01	C = 1e+00	C = 1e+01
0.518 ± 0.088	**0.580** ± 0.064	0.533 ± 0.083	0.569 ± 0.073	0.562 ± 0.078	0.544 ± 0.059

TABLE XXVIII
SVM WITH LINEAR KERNEL AT T2: F1 SCORE (MEAN ± STD.)

C = 1e-04	C = 1e-03	C = 1e-02	C = 1e-01	C = 1e+00	C = 1e+01
0.441 / 0.649	0.563 / 0.607	0.456 / 0.659	0.529 / 0.623	0.539 / 0.593	0.528 / 0.567

TABLE XXIX
SVM WITH LINEAR KERNEL AT T2: RECALL (MEAN) / PRECISION (MEAN)

C = 1e-04	C = 1e-03	C = 1e-02	C = 1e-01	C = 1e+00	C = 1e+01
0.607 ± 0.180	0.623 ± 0.127	0.664 ± 0.027	**0.669** ± 0.031	0.659 ± 0.032	0.657 ± 0.033

TABLE XXX
SVM WITH GAUSSIAN KERNEL AT T2: F1 SCORE (MEAN ± STD.)

C = 1e-04	C = 1e-03	C = 1e-02	C = 1e-01	C = 1e+00	C = 1e+01
0.465 / 0.530	0.826 / 0.521	0.908 / 0.524	0.911 / 0.530	0.892 / 0.523	0.886 / 0.523

TABLE XXXI
SVM WITH GAUSSIAN KERNEL AT T2: RECALL (MEAN) / PRECISION (MEAN)

	γ = 1e+00	γ = 1e+01	γ = 1e+02	γ = 1e+03
C = 1e-06	0.580 ± 0.068	0.588 ± 0.069	0.596 ± 0.075	**0.666** ± 0.065
C = 2e-06	0.578 ± 0.068	0.584 ± 0.069	0.630 ± 0.071	0.641 ± 0.066
C = 5e-06	0.586 ± 0.157	0.584 ± 0.058	0.600 ± 0.067	0.601 ± 0.065
C = 1e-05	0.591 ± 0.001	0.576 ± 0.054	0.586 ± 0.070	0.581 ± 0.070

TABLE XXXII
SVM+ WITH LINEAR KERNEL AT T2: F1 SCORE (MEAN ± STD.)

	γ = 1e+00	γ = 1e+01	γ = 1e+02	γ = 1e+03
C = 1e-06	0.566 / 0.602	0.584 / 0.610	0.767 / 0.600	0.879 / 0.593
C = 2e-06	0.556 / 0.600	0.572 / 0.604	0.770 / 0.609	0.795 / 0.605
C = 5e-06	0.579 / 0.558	0.568 / 0.547	0.614 / 0.594	0.686 / 0.601
C = 1e-05	0.595 / 0.500	0.555 / 0.525	0.578 / 0.575	0.587 / 0.584

TABLE XXXIII
SVM+ WITH LINEAR KERNEL AT T2: RECALL (MEAN) / PRECISION (MEAN)

	γ = 1e+00	γ = 1e+01	γ = 1e+02	γ = 1e+03
C = 1e-06	**0.669** ± 0.024	0.623 ± 0.212	0.667 ± 0.108	0.667 ± 0.080
C = 2e-06	0.288 ± 0.181	0.622 ± 0.120	0.667 ± 0.031	0.667 ± 0.029
C = 5e-06	0.638 ± 0.000	0.659 ± 0.000	0.629 ± 0.144	0.667 ± 0.242
C = 1e-05	0.647 ± 0.000	0.661 ± 0.000	0.560 ± 0.000	0.667 ± 0.000

TABLE XXXIV
SVM+ WITH GAUSSIAN KERNEL AT T2: F1 SCORE (MEAN ± STD.)

	γ = 1e+00	γ = 1e+01	γ = 1e+02	γ = 1e+03
C = 1e-06	0.931 / 0.523	0.456 / 0.418	0.560 / 0.508	0.860 / 0.530
C = 2e-06	0.309 / 0.485	0.737 / 0.536	0.620 / 0.525	0.140 / 0.522
C = 5e-06	0.878 / 0.500	0.888 / 0.500	0.604 / 0.518	0.140 / 0.491
C = 1e-05	0.879 / 0.500	0.901 / 0.500	0.548 / 0.500	0.140 / 0.500

TABLE XXXV
SVM+ WITH GAUSSIAN KERNEL AT T2: RECALL (MEAN) / PRECISION (MEAN)

D. T3 Complete Results

γ = 1e+00	γ = 1e+01	γ = 1e+02	γ = 1e+03
0.546 $\pm$ 0.063	0.551 $\pm$ 0.077	**0.553** $\pm$ 0.092	0.515 $\pm$ 0.092

TABLE XXXVI
LOGISTIC REGRESSION AT T3: F1 SCORE (MEAN $\pm$ STD.)

γ = 1e+00	γ = 1e+01	γ = 1e+02	γ = 1e+03
0.533 / 0.564	0.517 / 0.596	0.486 / 0.653	0.404 / 0.729

TABLE XXXVII
LOGISTIC REGRESSION AT T3: RECALL (MEAN) / PRECISION (MEAN)

C = 1e-04	C = 1e-03	C = 1e-02	C = 1e-01	C = 1e+00	C = 1e+01
0.572 $\pm$ 0.081	0.589 $\pm$ 0.075	0.540 $\pm$ 0.095	**0.596** $\pm$ 0.075	0.583 $\pm$ 0.076	0.556 $\pm$ 0.075

TABLE XXXVIII
SVM WITH LINEAR KERNEL AT T3: F1 SCORE (MEAN $\pm$ STD.)

C = 1e-04	C = 1e-03	C = 1e-02	C = 1e-01	C = 1e+00	C = 1e+01
0.513 / 0.659	0.547 / 0.648	0.445 / 0.712	0.556 / 0.652	0.566 / 0.609	0.550 / 0.570

TABLE XXXIX
SVM WITH LINEAR KERNEL AT T3: RECALL (MEAN) / PRECISION (MEAN)

C = 1e-04	C = 1e-03	C = 1e-02	C = 1e-01	C = 1e+00	C = 1e+01
0.594 $\pm$ 0.207	0.671 $\pm$ 0.028	**0.671** $\pm$ 0.026	0.670 $\pm$ 0.026	0.670 $\pm$ 0.025	0.669 $\pm$ 0.026

TABLE XL
SVM WITH GAUSSIAN KERNEL AT T3: F1 SCORE (MEAN $\pm$ STD.)

C = 1e-04	C = 1e-03	C = 1e-02	C = 1e-01	C = 1e+00	C = 1e+01
0.685 / 0.516	0.910 / 0.526	0.934 / 0.524	0.932 / 0.524	0.932 / 0.523	0.928 / 0.524

TABLE XLI
SVM WITH GAUSSIAN KERNEL AT T3: RECALL (MEAN) / PRECISION (MEAN)

	γ = 1e+00	γ = 1e+01	γ = 1e+02	γ = 1e+03
C = 1e-06	0.603 $\pm$ 0.064	0.593 $\pm$ 0.071	**0.636** $\pm$ 0.070	0.570 $\pm$ 0.073
C = 2e-06	0.632 $\pm$ 0.071	0.595 $\pm$ 0.070	0.634 $\pm$ 0.070	0.579 $\pm$ 0.064
C = 5e-06	0.616 $\pm$ 0.090	0.602 $\pm$ 0.063	0.606 $\pm$ 0.067	0.591 $\pm$ 0.069
C = 1e-05	0.605 $\pm$ 0.188	0.626 $\pm$ 0.163	0.594 $\pm$ 0.085	0.592 $\pm$ 0.075

TABLE XLII
SVM+ WITH LINEAR KERNEL AT T3: F1 SCORE (MEAN $\pm$ STD.)

	γ = 1e+00	γ = 1e+01	γ = 1e+02	γ = 1e+03
C = 1e-06	0.570 / 0.648	0.564 / 0.664	0.810 / 0.629	0.644 / 0.607
C = 2e-06	0.608 / 0.636	0.560 / 0.642	0.723 / 0.650	0.623 / 0.659
C = 5e-06	0.610 / 0.577	0.568 / 0.594	0.588 / 0.635	0.568 / 0.643
C = 1e-05	0.608 / 0.594	0.603 / 0.596	0.559 / 0.644	0.559 / 0.638

TABLE XLIII
SVM+ WITH LINEAR KERNEL AT T3: RECALL (MEAN) / PRECISION (MEAN)

	γ = 1e+00	γ = 1e+01	γ = 1e+02	γ = 1e+03
C = 1e-06	0.667 $\pm$ 0.000	0.667 $\pm$ 0.023	0.667 $\pm$ 0.026	0.665 $\pm$ 0.026
C = 2e-06	0.009 $\pm$ 0.000	**0.674** $\pm$ 0.000	0.667 $\pm$ 0.345	0.667 $\pm$ 0.298
C = 5e-06	0.670 $\pm$ 0.000	0.345 $\pm$ 0.000	0.667 $\pm$ 0.000	0.667 $\pm$ 0.000
C = 1e-05	0.671 $\pm$ 0.005	0.217 $\pm$ 0.000	0.667 $\pm$ 0.000	0.667 $\pm$ 0.000

TABLE XLIV
SVM+ WITH GAUSSIAN KERNEL AT T3: F1 SCORE (MEAN $\pm$ STD.)

	γ = 1e+00	γ = 1e+01	γ = 1e+02	γ = 1e+03
C = 1e-06	0.020 / 0.500	0.020 / 0.071	0.500 / 0.524	0.159 / 0.523
C = 2e-06	0.002 / 0.500	0.020 / 0.509	0.520 / 0.264	0.800 / 0.391
C = 5e-06	0.932 / 0.500	0.020 / 0.500	0.400 / 0.500	0.840 / 0.500
C = 1e-05	0.937 / 0.499	0.108 / 0.500	0.060 / 0.500	0.960 / 0.500

TABLE XLV
SVM+ WITH GAUSSIAN KERNEL AT T3: RECALL (MEAN) / PRECISION (MEAN)

E. T4 Complete Results

γ = 1e+00	γ = 1e+01	γ = 1e+02	γ = 1e+03
0.558 $\pm$ 0.070	**0.559** $\pm$ 0.082	0.549 $\pm$ 0.086	0.521 $\pm$ 0.082

TABLE XLVI
LOGISTIC REGRESSION AT T4: F1 SCORE (MEAN $\pm$ STD.)

γ = 1e+00	γ = 1e+01	γ = 1e+02	γ = 1e+03
0.553 / 0.566	0.525 / 0.604	0.481 / 0.650	0.411 / 0.728

TABLE XLVII
LOGISTIC REGRESSION AT T4: RECALL (MEAN) / PRECISION (MEAN)

C = 1e-04	C = 1e-03	C = 1e-02	C = 1e-01	C = 1e+00	C = 1e+01
0.575 $\pm$ 0.086	0.582 $\pm$ 0.080	0.532 $\pm$ 0.093	**0.591** $\pm$ 0.077	0.581 $\pm$ 0.068	0.563 $\pm$ 0.075

TABLE XLVIII
SVM WITH LINEAR KERNEL AT T4: F1 SCORE (MEAN $\pm$ STD.)

C = 1e-04	C = 1e-03	C = 1e-02	C = 1e-01	C = 1e+00	C = 1e+01
0.520 / 0.658	0.533 / 0.652	0.437 / 0.703	0.552 / 0.646	0.565 / 0.606	0.556 / 0.575

TABLE XLIX
SVM WITH LINEAR KERNEL AT T4: RECALL (MEAN) / PRECISION (MEAN)

C = 1e-04	C = 1e-03	C = 1e-02	C = 1e-01	C = 1e+00	C = 1e+01
0.555 $\pm$ 0.240	0.648 $\pm$ 0.122	0.672 $\pm$ 0.026	0.672 $\pm$ 0.027	0.672 $\pm$ 0.026	**0.673** $\pm$ 0.028

TABLE L
SVM WITH GAUSSIAN KERNEL AT T4: F1 SCORE (MEAN $\pm$ STD.)

C = 1e-04	C = 1e-03	C = 1e-02	C = 1e-01	C = 1e+00	C = 1e+01
0.652 / 0.485	0.897 / 0.530	0.937 / 0.525	0.937 / 0.525	0.938 / 0.524	0.937 / 0.525

TABLE LI
SVM WITH GAUSSIAN KERNEL AT T4: RECALL (MEAN) / PRECISION (MEAN)

	γ = 1e+00	γ = 1e+01	γ = 1e+02	γ = 1e+03
C = 1e-06	0.598 $\pm$ 0.069	0.577 $\pm$ 0.069	0.611 $\pm$ 0.074	0.574 $\pm$ 0.070
C = 2e-06	**0.622** $\pm$ 0.103	0.592 $\pm$ 0.072	0.612 $\pm$ 0.075	0.569 $\pm$ 0.070
C = 5e-06	0.603 $\pm$ 0.100	0.596 $\pm$ 0.090	0.594 $\pm$ 0.076	0.579 $\pm$ 0.075
C = 1e-05	0.594 $\pm$ 0.192	0.618 $\pm$ 0.144	0.591 $\pm$ 0.085	0.588 $\pm$ 0.076

TABLE LII
SVM+ WITH LINEAR KERNEL AT T4: F1 SCORE (MEAN $\pm$ STD.)

	γ = 1e+00	γ = 1e+01	γ = 1e+02	γ = 1e+03
C = 1e-06	0.561 / 0.650	0.527 / 0.656	0.726 / 0.619	0.497 / 0.598
C = 2e-06	0.597 / 0.658	0.554 / 0.645	0.645 / 0.652	0.582 / 0.658
C = 5e-06	0.592 / 0.613	0.555 / 0.615	0.562 / 0.639	0.541 / 0.643
C = 1e-05	0.594 / 0.607	0.588 / 0.645	0.554 / 0.632	0.550 / 0.641

TABLE LIII
SVM+ WITH LINEAR KERNEL AT T4: RECALL (MEAN) / PRECISION (MEAN)

	γ = 1e+00	γ = 1e+01	γ = 1e+02	γ = 1e+03
C = 1e-06	0.644 $\pm$ 0.128	0.644 $\pm$ 0.119	0.667 $\pm$ 0.027	0.667 $\pm$ 0.027
C = 2e-06	0.648 $\pm$ 0.124	**0.673** $\pm$ 0.027	0.667 $\pm$ 0.027	0.667 $\pm$ 0.187
C = 5e-06	0.672 $\pm$ 0.000	0.672 $\pm$ 0.004	0.628 $\pm$ 0.150	0.667 $\pm$ 0.231
C = 1e-05	0.672 $\pm$ 0.000	0.193 $\pm$ 0.000	0.571 $\pm$ 0.000	0.667 $\pm$ 0.000

TABLE LIV
SVM+ WITH GAUSSIAN KERNEL AT T4: F1 SCORE (MEAN $\pm$ STD.)

	γ = 1e+00	γ = 1e+01	γ = 1e+02	γ = 1e+03
C = 1e-06	0.890 / 0.514	0.888 / 0.520	0.540 / 0.525	0.580 / 0.524
C = 2e-06	0.895 / 0.512	0.937 / 0.525	0.539 / 0.525	0.580 / 0.376
C = 5e-06	0.937 / 0.500	0.937 / 0.500	0.566 / 0.508	0.580 / 0.475
C = 1e-05	0.937 / 0.500	0.186 / 0.500	0.624 / 0.500	0.580 / 0.500

TABLE LV
SVM+ WITH GAUSSIAN KERNEL AT T4: RECALL (MEAN) / PRECISION (MEAN)