Quiz: Gaussian Processes

STA414/2104 - Winter 2026

1. **Predictive Mean in Gaussian Process Regression**
   In Gaussian Process regression, let y_N be the observed targets with covariance C_N, and let k represent the covariance between the training points and a new test point. What is the predictive mean for the new point?

   (a) $k^\top y_N$
   (b) $C_N^{-1} k$
   (c) $k^\top C_N^{-1} y_N$
   (d) $c - k^\top C_N^{-1} k$

   Correct Answer: (c)

2. **Constructing Valid Kernels**
Given valid kernels $k_1(x, x')$ and $k_2(x, x')$, an arbitrary function $f(x)$, a positive semi-definite matrix A, and strictly positive constants $c_1, c_2 > 0$, which of the following combined operations is NOT guaranteed to yield a valid positive semi-definite kernel $k(x, x')$?

(a) $k(x, x') = \exp(c_1 k_1(x, x')) + f(x)k_2(x, x')f(x')$

(b) $k(x, x') = (k_1(x, x') + k_2(x, x'))^3 \cdot \exp(k_1(x, x'))$

(c) $k(x, x') = \exp(k_1(x, x') \cdot k_2(x, x') - x^\top A x')$

(d) $k(x, x') = c_1 k_1(x, x') \cdot f(x)k_2(x, x')f(x') + c_2$

Correct Answer: (c)
Rationale: While the product and exponentiation of kernels are valid, as is $x^\top A x'$ for a PSD matrix A, subtracting one valid kernel from another is not guaranteed to produce a positive semi-definite matrix, breaking the kernel's validity.

3. **Exact Inference Tractability**
Why is exact inference for Gaussian Process classification intractable, whereas it is analytically tractable for GP regression?

   (a) Because classification requires an infinite number of training points to form a decision boundary.

   (b) Because the kernel matrix becomes singular and cannot be inverted for discrete targets.

   (c) Because the prior over functions cannot be modeled as a Gaussian process for discrete outputs.

   (d) Because the likelihood function is non-Gaussian, resulting in a non-Gaussian posterior distribution.

   Correct Answer: (d)
   Rationale: Classification usually uses a non-linear link function (like a sigmoid) to squash the GP output, meaning the likelihood $p(y|a)$ is Bernoulli. A Gaussian prior multiplied by a Bernoulli likelihood does not yield a closed-form Gaussian posterior, making the integral intractable.

4. **Learning Hyperparameters**
How are the hyperparameters θ of a Gaussian Process's covariance function typically learned from the training data?

   (a) By minimizing the predictive variance across all test points.

   (b) By calculating the exact analytical solution by setting the gradient of the kernel to zero.

   (c) By performing Markov Chain Monte Carlo to find the maximum a posteriori (MAP) of the inputs.

   (d) By maximizing the log marginal likelihood $\log p(y|\theta)$ using gradient-based optimization.

   Correct Answer: (d)
   Rationale: The standard approach is to optimize the log marginal likelihood, which effectively tunes the hyperparameters to the data using gradient based optimization or grid search.

5. **The Kernel Trick**
What is the primary computational advantage of working directly with kernel functions (the "kernel trick") instead of explicitly calculating the feature map $\psi(x)$?

(a) We do not need to worry about the dimension of the feature space, allowing us to implicitly compute inner products in very high or infinite dimensions.

(b) It ensures that the regression model will never overfit the training data.

(c) It turns a non-linear classification problem directly into a linear regression problem.

(d) It makes the training time $O(1)$ regardless of the size of the dataset.

Correct Answer: (a)
Rationale: By working directly with the kernel, we can forget the feature map entirely; the dimension of the feature space does not matter anymore.

6. **Computational Complexity**
What is the primary computational bottleneck when training and making predictions with a standard Gaussian Process regression model on a dataset of N observations?

(a) Computing the $N \times N$ kernel matrix K, which scales as $O(N^2)$.

(b) Inverting the $N \times N$ covariance matrix C_N, which scales as $O(N^3)$.

(c) Optimizing the hyperparameters, which requires an exhaustive $O(2^N)$ search.

(d) Predicting the variance for a new test point, which scales as $O(N^4)$ per point.

Correct Answer: (b)
Rationale: To compute the predictive mean $k^\top C_N^{-1} y_N$ or the log marginal likelihood for hyperparameter learning, one must compute the inverse of the $N \times N$ covariance matrix C_N. This matrix inversion step has a computational complexity of $O(N^3)$, making standard Gaussian processes highly computationally expensive for very large datasets.