Kernel Selection & their parameters

1. Most used Kernels
   Gaussian Kernels (or RBF Kernel). It is because of having no prior knowledge about training data and gaussian kernels serves as a general-purpose kernel. But if we have large features dataset then linear or polynomial kernel is considered because to time complexity and this comes with less accuracy trade-off.

2. Comparison between Kernels [$x^T y$ - dot product between data point x and y]

- **Polynomial Kernel** ($K(x,y) = (\alpha x^T y + b)^d$)
  Parameters -
  α - slope
  b - intercept constant or bais
  d - degree

  Polynomial kernels are well suited for problems where all the training data is normalized.
  Three parameters to select which make it less popular.
  Linear kernel is one of the cases of Polynomial kernel where d = 1 and b = 0.

- **Radial basis function (RBF) Kernels**
  $$K(x,y) = exp(-\gamma ||x-y||^2)$$

  Gaussian Kernel ($K(x,y) = exp(-\frac{||x-y||^2}{2*\sigma^2})$
  Parameters -
  σ - sigma
  It is one of the radial basis function kernels (RBF) and is the most used kernel. Sigma plays a major role in the performance of the kernel, and should be carefully tuned to the problem at hand. If overestimated, the exponential will behave almost linearly and the higher-dimensional projection will start to lose its non-linear power. On the other hand, if underestimated, the function will lack regularization and the decision boundary will be highly sensitive to noise in training data.

  Laplacian Kernel $K(x,y) = exp(-\frac{||x-y||}{\sigma})$
  It is also one of the radial basis function kernels (RBF)
  Only one parameter to select hence this is also a reason to make it the most used one.

Sigmoid Kernel $K(x,y) = tanh(\alpha x^T y + b)$

Parameters -

α - slope

b - intercept constant

It is also known as multilayer perceptron (MLP) kernel. It comes from the neural network field and it is equivalent to 2-layer perceptron neural network.
A common value for alpha is 1/N, where N is the data dimension.

Note -
Kernel selection mostly depends upon the dataset and the kind of information we want to retrieve from the data. For examples - polynomial kernels gives the curvilinear hyperplane but gaussian can provide us with circular hyperplane also.
There are several reasons to select RBF Kernels over polynomials apart from less number of parameters. In RBF K_{ij} lies between 0 and 1 making it numerically more suitable over polynomial of which kernel values can go infinity or zero. The polynomial kernel is may suffer from numerical instability: when $x^T y + c < 1$, $K(x, y) = (x^T y + c)^d$ tends to zero with increasing d, whereas when $x^T y + c > 1$, $K(x, y)$ tends to infinity

RBF Kernel parameters
C (penalty factor) - C is the penalty parameters for misclassification. It controls the trade-off between errors of the SVM on the training data and margin maximization [Rychetsky (2001), page 82]. Higher the C, more is the penalty and hence hard margin. It has no intuitive meaning. But Mattera and Haykin (1999)(pages 226-227 in *Advances in Kernel Methods*) - "Let us suppose that the output values are in the range [0, *B*]. [...] a value of *C* about equal to *B* can be considered to be a robust choice."
So C is basically act as regularization parameters.

γ **-** Intuitively, the γ parameter defines how far the influence of a single training example reaches, with low values meaning ‘far’ and high values meaning ‘close’. The γ parameters can be seen as the inverse of the radius of influence of samples selected by the model as support vectors. The model is very sensitive to γ **.** if γ very large, the radius is the area of influence of support vectors only includes support vector itself and no amount of regularization with **C** will be able to prevent overfitting. When γ is very small, the model is too constrained and cannot capture the complexity or “shape” of the data. The region of influence of any selected support vector would include the whole training set. The resulting model will behave similarly to a linear model with a set of hyperplanes that separate the centers of high density of any pair of two classes. In practice it is interesting to simplify the decision function with a lower value of C so as to favor models that use less memory and are faster to predict.

Polynomial Kernel parameters

α (slope or scaling parameter) - The scaling parameter of the polynomial kernel is a convenient way of normalizing patterns without the need to modify the data itself.

b (bias or offset param) - It is basically used to compensate for feature vectors that are not centered around 0. Caret package always set b = 1 and do a grid search for α .

d (degree) - The most common degree is *d* = 2 (quadratic), since larger degree tend to overfit the data and cause numerical stability

Sigmoid Kernel parameters

Generally We go with RBF Kernel and search for best C and γ

By cross-validation and do a grid search method.

*For C and γ trying exponentially growing sequences is a practical method to identify good parameters (for example, C = 2^{-5} ,2^{-3} ,...,2^{15} , γ = 2^{-15} ,2^{-13} ,..., 2^{3}).

But Sometimes it's better to stick with linear kernel i.e not map the data

- When the number of features is very large than the data points. - in this case RBF do simply as good as linear so it's better to not map the data to save the time.
- Both the data points and features are very large (approx. range of 20K and 40K resp.)
- Number of data points is much much larger than features (approx 500K and 50 resp.)

References -

https://en.wikipedia.org/wiki/Polynomial_kernel
https://datascience.stackexchange.com/questions/1074/polynomial-kernel-parameters-in-svms
http://crsouza.com/2010/03/17/kernel-functions-for-machine-learning-applications/
* https://www.csie.ntu.edu.tw/~cjlin/papers/guide/guide.pdf
https://scikit-learn.org/stable/auto_examples/svm/plot_rbf_parameters.html
For see effect of γ
https://chrisalbon.com/machine_learning/support_vector_machines/svc_parameters_using_rbf_kernel/
This paper talks about various methods for kernel parameters selections apart from grid search
https://journals.sagepub.com/doi/pdf/10.1260/1748-3018.8.2.163