As technologies advance, more challenges are emerging. The variability and design rules become more extensive and more complex. Consequently, design tools have to solve increasingly complex problems, and these difficulties directly reflect on the cost of developing electronic device designs. One of the major challenges of nanoscale technologies is process variability, which affects the expected normal behavior of the cells, changing the delay and power consumption observed at nominal conditions. This expands the cell characterization to enclose now multiple corners, instead of the traditional fast, low and nominal corners, which can be a very time-consuming task. In recent studies, Machine Learning has been employed in several EDA applications to model non-linear relationships and to solve difficult problems. Therefore, this work evaluates machine learning regression algorithms as an alternative to the electrical simulation of logic cells. Four regression methods were investigated, namely Multiple Linear Regression (MLR), Support Vector Regression (SVR), Decision Tree (DT), and Random Forest (RF). As a case study, the models were trained to predict the propagation time and the energy consumption of the CMOS inverter circuit, using voltage, temperature, cell dimensions, and process variability as input variables. A comparative analysis is made for each variable between all four models in order to understand which one is the best regression model suited for the task. The best algorithm proved to be the Random Forest for all the predicted variables, with an RMSE of 10,3x10^-3 for predicting energy, of 3,2x10^-3 for predicting high-low propagation delay, and of 2,8x10^-3 for predicting low-high propagation delay. The prediction methodology presented in this paper will be extended to consider other circuits and technologies in the following steps of this project.

Olá, meu nome é Gabriel Jacinto, sou bolsista PIBIC pelo CNPQ, e, juntamente com a professora Cristina do ECL e o professor Mateus Grellert, pesquiso sobre a predição da caracterização elétrica de circuitos com algoritmos de Aprendizado de Máquina. Nessa apresentação irei dar uma breve introdução sobre o problema, indicar o que se objetiva com a solução, salientar os algoritmos utilizados, explicar os métodos e etapas que foram feitos para o treinamento desses algoritmos, discutir os resultados e concluir a apresentação.

No nível mais fundamental de todo equipamento eletrônico atual existe o transistor, o qual proporcionou uma revolução na computação. Hoje os transistores são muito mais pequenos do que essa imagem, sendo a tecnologia mais [4]recente de apenas 2 nm. Isso é menor do que o diâmetro do seu DNA (2,5 nm)[5], então sim absurdamente pequeno. [6]Por serem tão microscópicos, esses equipamentos estão à mercê de todo tipo de interferência, o que é um problema muito grande quando estamos [7]produzindo chips, pois não queremos lançar um equipamento que troque os valores de um bits, já que isso não só gera informações conflitantes, mas também afeta o consumo de energia[8] e atraso de resposta do aparelho. Para mitigar esse tipo de variabilidade, diversas [9]simulações são feitas durante a produção, só que elas podem demorar muito, e para nossa sociedade, tempo é dinheiro. Nesse sentido, com a enorme primavera de inovações criada por [10]algoritmos de Aprendizado de Máquina, podemos tomar proveito dessas técnicas para garantir que todo esse processo seja mais confiável, rápido e barato.

[a]O objetivo dessa pesquisa foi fazer um primeiro estudo em um circuito inversor CMOS para prever os seus atrasos e sua energia.

Na simulação do circuito inversor, calculam-se os valores para seu atraso, que é o tempo que o sinal demora para se propagar pelo circuito, e o valor da energia gasta. São essas características elétricas que queremos prever neste trabalho. Como estamos trabalhando com valores contínuos e não grupos (como se um animal é um cão ou um gato) estamos fazendo uma tarefa de regressão e não de classificação.

Cada algoritmo estará aprendendo a prever os valores de atraso de subida (tphl), o atraso de descida (tplh) e a energia (iint) do nosso inversor. Dessa forma, treinamos quatro algoritmos clássicos de regressão:

* [b]a regressão linear múltipla, onde nosso valor predito é descrito na pela seguinte equação, onde os tetas são os pesos de cada valor das variáveis de entrada. O que se deseja é encontrar os valores de cada peso que minimizem nossa medida de performance.
* [d]as árvores de decisão, que funciona criando uma estrutura em árvore com nós internos com testes que são feitos para prever a qual nó folha o valor será atribuído, retornando, em nosso caso, um valor contínuo.
* [e]a floresta aleatória, que calcula um grande número de árvores de decisão com profundidades diferentes e, em seguida, calcula a média de todas as previsões para estimar o valor.
* [c]e por fim a máquina de suporte de vetores, que dado um valor epsilon, o SVR calcula uma margem em torno do hiperplano da função de previsão, onde tentamos ajustar o máximo de observações possível entre a área definida pelas margens e reduzir os valores que são fora dessa.

[f]Para avaliar como eles performam em suas predições vamos usar a métrica conhecida como raiz quadrada do erro quadrado médio.[h]Um detalhe importante que tivemos que ajustar em nossos dados foi a escala das nossas variáveis, já que a magnitude delas é muito pequena (estamos falando de uma escala menor que nano). Logo, nós aplicamos a seguinte fórmula, conhecida como MinMaxScaler, em todos os valores para que eles ficassem entre o intervalo zero e um.

[g]Aqui podemos ver em mais detalhes todas as nossas variáveis de entrada junto com seus valores e nossas variáveis a serem preditas. No final da simulação e após remover alguns valores discrepantes, tivemos um conjunto de 120.954 dados, sendo 50% para o treinamento do algoritmo, 25% para a validação e 25% para o teste dele a fim de se evitar o overfit e enviesamento.

Depois de todo esse workflow, a distribuição de nossas variáveis de predição ficou da seguinte forma, nós vamos usar essa distribuição para visualizar de forma mais intuitiva as predições feitas por cada algoritmo.

[i]Como podemos ver, para o TPHL, é quase impossível ver algum erro nas curvas preditas pelas árvore de decisão e pela floresta aleatória e isso fica ainda mais claro quando analisamos os resultados do erro quadrático médio, que nos diz que o melhor algoritmo foi realmente o florestas aleatórias e o pior a regressão linear múltipla. Seguindo para o TPLH, nossa curva característica aqui em laranja, podemos ver que o mesmo padrão se repete, com o melhor algoritmo sendo a floresta aleatória e o pior a regressão linear múltipla. Por fim temos a energia, com sua curva aqui em verde e mais achatada, e, sem surpresa nenhuma, a floresta aleatória ganha novamente dos outros algoritmos. É interessante notar a pequena diferença entre os erros dela e da árvore de decisão, o que faz sentido, já que a floresta em si é composta por diversas árvores.

[j]Nessa conjuntura, podemos ver que o nosso algoritmo campeão foi o florestas aleatórias, o que já era até esperado, já que é um dos algoritmos mais poderosos. Podemos concluir que algoritmos de machine learning são capazes sim de prever a caracterização elétrica de circuitos. Contudo, esses algoritmos ainda são relativamente simples quando olhamos para o cenário atual de pesquisa em aprendizagem de máquina, então à medida que formos pegando circuitos cada vez mais complexos para fazer a predição, muito provavelmente iremos usar de técnicas mais robustas como redes neurais. Tudo isso com o intuito de acelerarmos o processo de simulação de circuitos. [l]Por fim, com os resultados desse trabalho, disponibilizou-se o melhor modelo em uma aplicação web feita com a biblioteca Streamlit da linguagem python. Obrigado pela atenção!